SINCLAIR QL WORLD

OCTOBER 1991 £1.95

M.C.M
Quality Editorial

9 770951 933016 10

A QL cousin:
The One-Per-Desk

Software File:
The Painter
update

Get inside your QL's head:
SYSTEMATIC MACHINE CODE PROGRAMMING

DJC

Dilwyn Jones Computing

41 Bro Emrys, Tal-y-Bont,
Bangor, Gwynedd LL57 3YT
Tel: Bangor (0248) 354023

DJC HOPES TO HAVE A STAND AT THESE QUANTA QL WORKSHOPS IN THE NEAR FUTURE:
SEPTEMBER 22ND (PORTSMOUTH)
SEPTEMBER 28TH (BIRMINGHAM)
OCTOBER 12/13TH (NOTTINGHAM)
NOVEMBER 17TH (BRISTOL)
WE ALSO HAVE A STAND AT SOME "ALL FORMATS" SHOWS FROM TIME TO TIME - CHECK WITH US.

THIS MONTH:
* Lower prices on 3.5 inch floppy disks
* 6 new QL software packages
* Posso Disk Boxes back in stock
* Disk Box Subject Dividers back in stock
* Programs on mdv's priced separately
* New catalogue ready - send SAE for a copy

It has more details about our products

NEW QL SOFTWARE

PROGS (Professional & Graphical Software)
High quality 100% machine code QL software by PROGS, from Belgium. On floppy disk only. Ask for more information.

THE PAINTER **£25.00**
Superb painting program for the QL (512k)

THE CLIPART **£12.00**
3 disks full of QL clipart. (128k)

DATA DESIGN version 2 **£50.00**
Fast, flexible database. Programmable from BASIC or assembly language (512k)

QRACTAL ***£20.00***
Fractals program, 100% machine code, displays Mandelbrot or Julia sets. (512k)

ROB ROY QL BARGAIN SOFTWARE PACK
On disk only £10.00
on 3 microdrive cartridges £14.00
Reviewed in QL World August 1991, this package features a set of 4 Rob Roy Software programs (128k)
INKWELL DELUXE - 100's of print styles from Epson compatible 9 pin dot matrix printers. Font editor + 20 extra fonts
INKWELL DELUXE TYPER - Typewriter style extension to Inkwell Deluxe
CUEWELL 1 + CUEWELL 2-front end utilities

ADDRESS LABEL PRINTER by J.H. Ree **£15.00**
On microdrive cartridge **£15.00**
A really simple to use, yet highly useful utility for storing names, addresses and phone numbers and for printing address labels, etc, with several novel facilities - ask for details. (384k, RAMdisk required)
100 ADDRESS LABELS on printer roll for use with Address Label Printer **£2.00**
add 50p postage per roll (max £2.50)

REMIND ME ***by Joe Haftke £12.00***
On microdrive cartridge **£12.00**
Fast, compact, easy to use dates reminder program, for birthdays, bills, etc. (128k)

QL. GENEALOGIST ***by Chris Boutal £19.50***
128k version on cartridge **£19.50**
Expanded version on 2 cartridges **£21.50**
Our best selling family history and family trees program. (128k/384k versions)

SUPER DISK LABELLER **£10.00**
On cartridges if required **£12.00**
Print labels for floppy disks, can show the filenames on disks in columns. (256k)
ROLL 100 DISK LABELS for above **£2.50**
add 50p postage per roll of labels

SUPER DISK INDEX ***Imre Dominik £12.00***
On microdrive cartridge **£12.00**
Catalogue and index your disks and/or cartridges in one master database. Catalogue/sort/list/search for files, helps you get really organised (256k)

DISCOVER ***by Dave Walker £20.00***
QL/PC/ST disk file transfer (256k, disk only)

MULTI DISCOVER ***by Dave Walker £30.00***
As discover, also transfers to BBC DFS/ADFS, CPM (many formats) and UNIX CPIO formats (256k, disk only)

TEXTIDY ***by Dave Walker £15.00***
Text file conversion utility, e.g. for use with Discover above. (256k, disk only)

QUICK POSTERS ***by Dilwyn Jones £10.00***
on 2 microdrive cartridges **£12.00**
Star printers text poster maker (128k)

BASIC REPORTER ***by Dilwyn Jones £10.00***
on microdrive cartridge **£12.00**
Super BASIC programming aid (128k)

SCREEN ECONOMISER ***by G. Estournet £10.00***
on microdrive cartridge **£12.00**
Screen protection and blanking utility - turns off the QL display if no key is pressed for set number of minutes.

TRANS 24 ***by Ralf Reköndt £10.00***
on microdrive cartridge **£12.00**
9 - 24 pin printer graphics dumps conversion utility (128k)

FILES 2 ***by Peter Jefferies £12.00***
on microdrive cartridge **£12.00**
Comprehensive popup files utility. (128k)

SPELLBOUND ***by Peter Jefferies £30.00***
on microdrive cartridge **£30.00**
Check spelling as you type, 30,000 word dictionary, expandable by the user. (256k)

SPELLBOUND SPECIAL EDITION ***£50.00***
not available on microdrive cartridge
Larger dictionary, turns back on automatically at start of new words. (512k)

UPGRADE, SPELLBOUND TO SPELLBOUND S.E. ***£30***
Return Spellbound master disk/cartridge

FLASHBACK ***by Peter Jefferies £25.00***
on microdrive cartridge **£25.00**
Excellent fast, versatile database (256k)

FLASHBACK SPECIAL EDITION ***£40.00***
not available on microdrive cartridge
Enhanced version of Flashback - new commands, report generator, etc. (256k)

TASKMASTER ***by Peter Jefferies £25.00***
on microdrive cartridge **£25.00**
Front end utility for the QL (384k)
Hard disk users - ask about the version fo Taskmaster for hard disks

WINBACK ***by Norman Dunbar £25.00***
Miracle hard disk backup utility. (256k, Toolkit 2, available on disk only)

THE GOPHER ***by Norman Dunbar £12.00***
on microdrive cartridge **£12.00**
File finder utility. WIN or FLP (128k)

QUESTION MASTER ***by C.B. Storey £10.00***
on 2 microdrive cartridges **£12.00**
Question/answers quiz and revision system. Quiz editor included (128k)
General Knowledge quiz **£5.00**
Classical music quiz **£5.00**
QLestions - quiz on the QL itself! **£5.00**

FILEMASTER ***by Joe Haftke £12.00***
on microdrive cartridge **£12.00**
File care utility, directory display in columns, disk labelling. (512k, RAMdisk)

HOME BUDGET ***by Joe Haftke £20.00***
on microdrive cartridge **£20.00**
Personal income tax calculator (UK) plus domestic bills and accounts system (128k)

VISION MIXER 1 ***by Dilwyn Jones £10.00***
on 2 microdrive cartridges **£12.00**
Screen pictures video effects display, as reviewed in QL World Jan 1991 (384k)
VISION MIXER PLUS ***by Joe Haftke £22.50***
available on disk only
More features than version 1, even easier to use. (256k)
Upgrade VISION MIXER 1 TO VISION MIXER PLUS - *(quote serial number)* ***£14.50***
PICTUREMASTER ***by Joe Haftke £15.00***
available on disk only (256k)
Screen making utility for Vision Mixers.
VISION MIXER PLUS + PICTUREMASTER - both together, save 20% ***£30.00***
PICTUREMASTER + the upgrade ***£24.00***

COCKTAILS WAITER ***by Imre Dominik £10.00***
on 2 microdrive cartridges **£12.00**
Cocktail drinks recipes (over 400 supplied!) database. (256k)
MIX 2 - hundreds of extra recipes **£5.00**
MIX 3 - even more recipes, includes many non-alcoholic recipes **£5.00**
MIX 2 or 3 on microdrive cartridge **£6.00**

DI-REN PRODUCTS
FLEET TACTICAL COMMAND II **£49.95**
on microdrive cartridge **£56.95**
DATA PRINT UTILITY for FTC II **£9.95**
FLEET TACTICAL COMMAND I **£29.95**
THE FUGITIVE text adventure **£9.95**
QL NETWORK PROVER hardware **£3.50**
MINI PROCESS CONTROLLER hardware **£59.95**
6 relay switches for the QL to control external equipment via the serial port.
MPC UTILITY software **£9.95**
Control program for process controller. Includes tookit of BASIC extensions for writing your own controller software.
Leads and PSU's for the MPC also stocked.
Send SAE or international reply coupon for a copy of the DI-REN catalogue.

SUNDRY SUPPLIES

Brand new microdrive cartridges	£2.50	100 address labels on roll, 3.5" wide	£2.00
20 or more	£2.25	Disk storage boxes for 3.5" disks:	
Roll 100 cartridge labels	£2.00	10 disks (Slimpak boxes)	£1.20
3.5" DSDD unbranded disks + labels	£0.45	40 disks, clear, lockable hinged lid	£5.00
20 or more	£0.40	80 disks, clear locakable hinged lid	£7.00
3.5" Sony branded disks + labels	£1.00	160 disks, POSSO box, drawer style	£17.00
pack of 10	£9.00	Disk box subject dividers, headings on one side, blank on the	
5.25" DSDD unbranded disks + labels	£0.35	other, dry wipe, per set of 20	only £3.00
Extra disk labels, roll 100	£2.00	Monitor stand, tilt and swivel, holds monitors with screens up to	
100 disk labels on printer roll	£2.50	14"	£15.50

Printer ribbons - we can obtain most types. These are normally held in stock, price £4.00 each + 50p postage per ribbon (max. postage £2.50)
GLP/GLP 2 & M1009/M1109, Brother HR5, Star LC10, LC24-10, LC24-200, Citizen 120D, FX/MX/RX80 etc. Juki 6100, Canon PW1080/Kaga KP810, Shinwa CPA80, Amstrad PCW8256/512/Seikosha SP1000/Sinclair printer.

WE WOULD LIKE TO THANK OUR CUSTOMERS DURING OUR FIRST YEAR OF ADVERTISING IN QL WORLD, AND WE LOOK FORWARD TO BEING ABLE TO SERVE YOU AS LONG AS THE QL MARKET IS STILL VIABLE.

SOFTWARE DISCOUNTS
Buy 2 programs together **CLAIM 5% OFF**
Buy 3+ programs together **CLAIM 10% OFF**
Offer applies to software only.
Vision Mixer and PictureMaster packages together described above count as ONE program for the purpose of this discount. This is not as stated last month - we made a mistake. Sorry!

SOFTWARE POSTAGE: UK - Software sent post-free in UK except Fleet Tactical Command, for which you should add £2.50. ABROAD - add £1.00 per program for postage and packing. For Fleet Tactical Command, you should add £3.50. SUPPLIES POSTAGE: UK - add £2.50 postage for disks, stands, boxes, DI-REN hardware and any other bulky items. Rates for labels and ribbons listed above. Blank microdrives sent post free in UK. ABROAD - add standard rate £3.50 postage, except for airmail outside Europe, where you should add 10% to order value, minimum air mail charge £3.50. PAYMENT - In British currency (Pounds Sterling) only, please. Examples of valid forms of payment are Sterling cheque drawn on UK bank or building society, Eurocheque, Draft, International Postal or Money order - your bank or Post Office may be able to advise you on how to pay u s in British currency. Make cheques etc. payable to DILWYN JONES COMPUTING. In cases of difficulty contact us first to arrange details. We hope to have credit card payment facilities before the end of the year, please check with us by phone before ordering if you wish to pay by this means to see when this service will be available. Goods remain the property of Dilwyn Jones Computing until paid for in full. Send an SAE or international reply coupon for further details of any program, or for a copy of our new catalogue.

SINCLAIR QL WORLD

Editor
Helen Armstrong

Production Controller
Jayne Penfold

Designer
Jeff Gurney

Advertising Manager
Jason Newman

Magazine Services
Yvonne Taylor

Advertising Production
Michelle Evans

Group Advertising Manager
Lynda Elliott

Group Editor
John Taylor

Publishing Director
Wendy Palmer

Group Deputy Managing Director
Ray Lewis

Group Managing Director
Peter Welham

Sinclair QL World
Panini House
116-120 Goswell Road
London EC1V 7QD
Telephone 071-490 7161
ISSN 026806X

Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write the The Editor, Open Channel, Trouble Shooter, or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.
Back issues are available from the publisher price £2 U.K., £2.75 Europe. Overseas rates on request.
Published by Maxwell Consumer Magazines, A Division of MCPC Ltd., Sinclair QL World is distributed by IPC Market force, King's Reach Tower, Stamford Street, London SE1 9LS.
Subscription information from: MCM Subscription Dept, Lazahold Ltd., PO Box 10, Roper St, Pallion Ind. Est., Sunderland SR4 4SN. Tel: 091 510 2290
UK: £21.00, Europe: £32.00,
Rest of the World: £38.00.

Typesetting by Ford Graphics, 8-10 Whitsbury Road, Fordingbridge, Hampshire. SP6 1BR. Tel: (0425) 655657
Printed and bound by BPCC, Colchester. Covers printed by Spottiswoode Ballantyne, Colchester.
Sinclair QL World is published on the third Thursday preceding cover date.

© COPYRIGHT
SINCLAIR QL WORLD – 1991

CONTENTS

OCTOBER 1991

4 TROUBLESHOOTER ● QL roms and Atari
13 QL SCENE ● Price shift at CGH
14 OPEN CHANNEL ● Review success
16 SOFTWARE FILE ● The Gopher
19 QL SCENE ● New Quanta Workshop
22 SOFTWARE FILE ● The Painter
24 A QL COUSIN ● The One-Per-Desk
25 THE NEW USER GUIDE ● Part 8 – Data arrays
29 SOFTWARE FILE ● Disk Labeller
32 SYSTEMATIC MACHINE CODE PROGRAMMING ● Part 1
37 DIY TOOLKIT ● Task Commander
41 MICRODRIVE EXCHANGE
42 DBQL ● PART 4
48 CLUB ACCESS
50 INSTANT ACCESS
BACK COVER – SUBSCRIPTION INFORMATION

NEXT MONTH

PROSPERO PASCAL
An alternative language for the portable programmer.

ONE MAN'S SYSTEM
Getting into the swim with Archive.

TROUBLE

A PROBLE

Brian Davies reflects on screen shots and emulators.

The process of transferring files from floppy to hard disk caused caused me a temporary loss of *text*[87], which did not want to run from the hard disk. The RECON routine had been run, and the device changed to WIN1_, but the program wasn't getting past the start of displaying it as text area when loaded. The way out of the problem was provided by software[87], with the suggestion to try deleting the file CONFIGURE_C87 from the hard disk. When this had been done, the program started normally and it was possible then to alter the Text and Storage Device settings to WIN1_ and save a new CONFIGURE_C87 file.

While my reservations about the implementation of the directory structure on the Miracle hard disk still remain, there is no doubt that the space and speed brought with it by the hard disk make QL life much more pleasant. The freedom to be able to copy a seemingly endless string of large files onto the hard disk, without coming near running out of space, relieves a considerable pressure from the user who has been having to consider every move very carefully previously.

What's your rom?

Do you know what version your Qdos rom chips are? (Type PRINT VER$, then enter, and the version will be printed on the screen.)

Many users possibly do not have the faintest idea of their version, and are not really interested. In some senses unfortunately, programmers *are* interested, and they tend to use the 'best' version – JS. This is only to be expected; if the JS solves some faults in the computer, it is reasonable to use it. What is sometimes forgotten though, is that many users do not have this version. The result may be that the program, which works fine for the programmer, causes trouble for the user. This has happened with three programs I have used recently, and two of them were effectively unusable on my JM. My feeling about this is rather the same as with cars – petrol should be designed to suit the car, not the car to suit the petrol, and it is distinctly upsetting to have a relatively expensive investment rendered almost useless by changes in its diet. Can't programmers test on more than one version of rom? QLs are not that expensive.

Version 2.08 of *Discopy* from Qfile is now out, with a change having been made to the boot procedure to prevent the 'in use' message, noted on a JM QL when version 2.06 was reviewed.

Printing

Once you get a new printer, you find all sorts of things to try it out on. The higher - quality output of ink jet and laser printers (compared to dot-matrix types) is clear to see when text is printed from a word-processing program, but the advantage is less obvious to see when printing graphics, from a DTP program for instance. Prints on a Hewlett-Packard DeskJet and Epson GQ-5000 laser from *Professional Publisher* seem better in appearance, but mainly because the paper gets less damaged by the print process (no impact from pins) and these printers use cut-sheet paper, which may be of better surface appearance than the continuous paper often used in DMP printers. So the quality of the image is essentially no better, being determined, to a large extent, by the printer-driver and printer-emulation mode used, which has to be 'Epson-compatible' (that is, FX80-compatible, a rather ancient and basic standard for both text and graphics). If my figures are correct, the resolution of the laser in its native mode is about four times better than that of the FX80.

The DeskJet (using an FX80 emulation cartridge) also produced some oddities in text. For example, lots of 's' characters looked more like 'z'. The laser won't (so far) print a full page on one sheet of paper; it spreads one ProPub page over three sheets. This problem is somewhat mystifying, and comments on it would be appreciated. The impression is given (by the printer) that it has run out of memory, part-way through a print. It says 'buffer full'. Bearing in mind that the page being printed is on a disk file of less than 100 KB, and the same printer happily handles files of around 500 KB from other sources, lack of memory *per se* can't be the problem. The printer has 2 MB of ram anyway. What does seem to be involved is the input buffer of the printer, which can be no larger than 51 KB.

In the picture

The quality of photographic printing from shops and mail - order companies is nothing to boast about, but it usually seems that the negatives and transparencies are good enough; it's the prints that are poor, possibly mainly because of the inadequacies of automatic focusing devices used in the print process. If you want to take shots of your QL screen, you don't really have to worry about your camera being a cheap one, since the advantages of a good camera tend to get thrown away by the printers. Read the part of the *QL User Guide* devoted to Easel for advice. Basically, a shutter speed of no less than about 1/4 second is advised for TV screens. This is to ensure that an even, steady picture is obtained. The TV screen picture takes about 1/25 of a second to 'draw', so the exposure needs to be greater than this. With a 100 ASA film, an aperture of f5.6 is recommended. To reduce screen distortion, it is preferable to use a lens of 100 mm or so focal length.

My own recent shots of a colour monitor screen were taken using cheap, mail-order 100 ASA film, a shutter speed of 1/5 or 1/2 second, an aperture of f5.6 and a 135 mm lens. On reflection, the shutter speed should have been higher, say, 1/15 second, but my camera offers only 1/25 in this range. That is rather too close to the screen 'painting' rate, but the shutter may have got a bit slow in its old age, and could in practice be about 1/20 and suitable for the job. The 1/3-second exposures were wasted. The telephoto lens does necessitate the camera being 2 metres away from the screen, but it reduces distortion. For one display, the camera tripod was backed against a cupboard door, and it was essential to have a viewfinder which allowed looking down on the image, since there was no room to get behind the camera and look through the usual viewfinder. Use a cable release for the shutter, to reduce the risk of shake. Make sure the room is dark, and that there are no reflections on the screen. Adjust the display controls to make the image sharp and free of glare, but make sure no important colours are lost by turning brightness of contrast controls too far.

Boots charge £4.59 to do one-day processing of 24-exposure 35 mm film – not cheap, but acceptable if you are in a hurry. The prints must be glossy for the printers who have to transfer the image to

SHOOTER

M S O L V E D

the printed page. Don't bank on prints coming out well. The last set of mine had several chopped at the edges, and out-of-focus, despite the transparencies looking quite reasonable. The transparencies themselves are likely to be preferable, anyway, because they are much sharper and allow the printers more flexibility. While we do use black and white screen dumps (a graphics transfer of the screen image to a printer) to illustrate program reviews, and they are satisfactory in many cases, colour can be important with some programs and it does look more attractive on a magazine page.

Brief encounter: Atari QL emulator

QL World has been asking for owners of an Atari ST with the QL emulator fitted to step forward and let the rest of us know something about this combination. One (very satisfied) owner of the combination is Alf Kendall, and he was kind enough to bring the system round and demonstrate it to me. Here are a few notes based on watching the system demonstrated for most of one day.

The basic computer is an Atari Mega 2, with 2 MB (expandable to 4 MB) of ram memory and an 8 MHz 16-bit 68008 processor chip, and one 3½ in floppy drive.

The computer casing is about the same width as the Phillips CM 8833 colour monitor which sits on top of it. There is another similar-sized box underneath, in which resides a 30 MB hard disc drive. There is room for another hard disc, and – with some persuasion – a floppy drive, but these are not presently fitted to Alf's system. He has another 3½ in drive attached externally. While the units mentioned so far make up quite a useful system, there are a trio of extra hardware items which make it really interesting.

An accelerator board boosts the CPU speed to 16 MHz. Two other internally-mounted boards give the capability to emulate MS-DOS – that is, a PC – and Qdos – a QL. So, you effectively have three computers in one. You can also buy a Macintosh emulator, to make that four computers. It may sound a bit of a cobbled-up thing, but it really does work, and work well. The hard disk (on Alf's system) is partitioned into three sections: Atari dos, MS-DOS and Qdos. The MS-DOS emulator says it has a '16-bit 8 MHz 80286' CPU, and it certainly behaves as though that's true. It appears to run PC programs at a similar speed to my 10 MHz 286 machine, and the disk-access speed is nearly as high as on my machine (which has a 0.5 MB disk cache to assist it). Using the Norton Utilities System Information benchmark routine, the overall speed was about eight times that of the reference PC/XT. The two floppy drives and one partition of the hard drive are usable with MS-DOS. Various standard programs (eg *WordPerfect 5.0, Flight Simulator 3.0, Lotus Manuscript, Psion Abacus* and *Quill)* ran without a problem. The screen display was of only CGA standard, but it does look likely that there is a way of obtaining EGA; this is something to be checked later.

The Qdos emulator has been around for three to four years and is well debugged. The only program Alf quoted as not running properly is the configuration routine for *Turbo-Plus-Quill.* We ran *Taskmaster, Flashback, Perfection, text*^{87}*, Archive, Professional Publisher* and various other standard QL programs. The absence of trouble was impressive. Equally impressive was the speed, generally quoted as about three times that of basic QL, for the basic 8 MHz Atari. Even Quill ceases to be a real drag on this combination computer. Not surprisingly, Perfection zooms along. Going up and down on a ProPub page with the cursor checked out as being six times as fast as on my JM QL, and the screen movement was much smoother. The speed- up factor is very likely overall more like six than three because the 16 Mhz accelerator board improves QL performance as well as Atari (but apparently doesn't do anything for the MS-DOS emulator). Alf uses a small routine which allows him to select different sub-directories, before switching (with Taskmaster) from one program to another. This deals with the problem experienced with the Miracle hard drive on the QL, where you can find the program you have switched to working from the wrong data sub-directory (this routine can be used on the normal QL, too). One hard disk partition and both floppy drives are accessible to Qdos.

Readers' letters

Vladimir Sams reports from Belgrade that he has now got his 'home' QL running, after what appeared to be display-induced failure. The MC1377P chip (IC28) was replaced by a local repairer. He mentioned that the Washington DC QL group in the USA has details of a buffered display cable, to prevent displays killing chips through the unbuffered QL interface. Sams now has another problem – how to get two battery-backed clocks (bought in the USA) working. Does anyone know of 'standard' problems with such clocks?

P Land has got his printer spacing correctly (it previously double-spaced with Quill and Abacus but not with SuperBasic programs). His problem was quite common, but no easier to solve, at first, just because it was a simple one. It was a question of what code(s) the printer was set to expect, to move one line space. Typical dot-matrix printers have an internal switch setting which allows them to Line Feed, or not to Line Feed, when a Carriage Return command is received from a computer. If the printer is set to add a Line Feed to every Carriage Return automatically, and the computer program is set to do the same, the result is double-spacing instead of single. You can set the Quill printer-driver to give an EOL (end of line) command to the printer of CR + LF, or just CR. To match just a CR, the printer itself has to respond to the pressing of the ENTER key with a Carriage Return (the print head returns to its leftmost position) and a single Line Feed. Land set the Quill printer-driver to give only CR when the ENTER key is pressed; this is an easier solution than getting into the bowels of the printer to reset a DIP switch, but fixing the problem this way might result in trouble printing from other programs in future, and there is no guarantee that the other programmers will have provided the facility to set the EOL code(s).

Since writing comments about using fax paper rolls for the Serial 8056 printer in a previous issue, a fax machine has been added to my own collection, making me more familiar with the subject. The Amstrad FX 9600AT can accommodate both 210- and 216 mm -wide rolls, with the aid of two spacing pieces supplied as standard with it. The thermal paper rolls are 30 m in length and cost £3.60 + VAT (210 mm) from my local computer shop. The more usual retail price appears to be about £5–6 per roll. A 30 m roll is 5.5 cm in diameter.

Perhaps equally important is the internal diameter of the core of the roll, which is 2.5 cm in the case of the Amstrad fax machine. Letters received so far seem to imply that only roll width matters, but a supplier of fax paper (see Information box below) has been kind enough to write in response to the note about **R Thompson,** and the letter suggests that ½ in is the required core diameter for the 8056. If you are having trouble getting paper for your 8056, and decide to contact this supplier, it might save time to inform them of the internal core diameter of the 'proper' rolls, and supply a sample piece of paper.

As an aside to this matter, the Amstrad machine has a din connector on the back 'for passing scanned images to the computer'. This was news to me, and I suspect that there are no suitable interfaces available for processing the fax-machine output so that it could be accepted by a graphics program. Does anyone have experience of feeding images scanned by a fax machine into a computer?

Mike Jackman had not (as of late April) managed to find a solution to his problem with *TechniKit/TechniQL,* from **Talent Computer Systems.** He successfully uses the Plot module of the program on his QL to produce output from a Penman plotter, but cannot get QL screens dumped from the Screen Dump module. The program 'does not run at all' on his Thor. Can anyone shed any light on this subject?

The only well-known QL supplier left that supplies a wide range of software written by people who have since left the QL scene is **TK Computerware.** It is good to have a supplier willing to deal in obsolete items, but it is almost inevitable that the supplier will come in for criticism from customers who feel they are not getting the usual speedy service, when they ask for help with old programs. When a program is actually produced by a supplier, the customer has good reason to hop up and down if complaints are not dealt with quickly, but there is nothing much a supplier can do when it cannot get a good response to enquiries about old programs, from other suppliers or programmers who may no longer be interested in QL matters.

Some users seem to develop a love-hate relationship with all or part of their Computer system, and **P H Tanner** seems to fall into this category where Minerva is concerned. He emphasises that he thinks that the product is '24 carat', but is far less complimentary about the support for it. In his running battle to get his two QLs talking to each other and doing (Forth-language) work he wants to get on with, he has at least achieved one success, in contradiction of 'what the experts have told him'. It has long been my understanding that a decent network between two QLs is not possible unless both of them are fitted with *Toolkit II* , on rom. Tanner says he now has a good network, without TKII on rom being in both computers. He doesn't say whether he is using TKII at all, however, and it may be worth pointing out for the benefit of other would-be networkers that the basic QL network commands do work, after a fashion, without any assistance from any toolkit. The troubles are that: the network set up in this way is not reliable; the available commands are very limited; and the QL-network hardware varies in its reliability from computer to computer. Another step in the right direction for him is that he has found a way of getting *QFlash* to work with Minerva.

Italy has a fair number of QL enthusiasts, one of whom is **Francesco Querini,** who has written with some comments on possible future hardware developments. He has designed various units for **SPEM,** an Italian company which has been producing hardware for the QL for several years now. One idea he mentioned is an advanced graphics driver board. Undoubtedly, many users would like to have a sharper screen picture, and that is now possible on the Atari ST with QL emulator, but not on the QL itself. With the greater speed and memory now available from the *Gold Card* , it would be nice to do more justice to programs such as Professional Publisher, but would the types of display most of us have (eg Philips CM8833, Microvitec Cub 653) be able to give a sharper picture?

INFORMATION
Fax paper for Serial 8056: ACD Morgan, Sensitised Coatings Ltd, Bergen Way, North Lynn Industrial Estate, King's Lynn, Norfolk PE30 2JL. Tel: (0553) 760377.

CARE ELECTRONICS

QL SUPERTOOL KIT II by Tony Tebby

Over 118 Commands:- Full Screen Editor, Key Define Print Using, Last Line Recall, Altkey, Job Control, File Handling, Default Directories, Extended Network,
16k Eprom Cartridge Version ..... @£23.50d
Configurable Version on Microdrive ..... @£23.50d

MIRACLE SYSTEMS PRODUCTS

OK Disc Interface (Inc Tool Kit II) ..... @£99.64b
QL Centronics Printer Interface ..... @£29.61d
QL Expanderam 512K Thrucard ..... @£99.64b

QL HARDWARE

Single 3.5" Disc Drive & (Own PSU) ..... @£105.75a
Dual 3.5" Disc Drive & (Own PSU) ..... @£176.25a
3.5" DS/DD Discs - 10 off ..... @£9.40c
QL Keyboard Membrane ..... @£11.75d
Care Eprom Cartridges each ..... @£6.11c
ULA CHip ZX8301 ..... @£15.98c

SOFTWARE 87 (State MDV or Disc)

TEXT87 (Budget Version) ..... @£45.98d
FOUNTED 89 ..... @£15.00c
FOUNTEXT 88 ..... @£25.00c
TEXT 87/FOUNTED 89/FOUNTEXT 88 ..... @94.94b
248B PRINTER DRIVER ..... @£15.00c
Upgrade to Text 87 V.3.00. Return old copy together with ..... @£25.00d

MONITORS (Price including lead)

Philips BM7502 Green Hi-Res £99.64a

QPAC II by Tony Tebby

MAKES YOUR QL TRULY MULTI - TASKING
QPAC II Main menu windows adjust automatically to size. Files, directory, view, print, delete, backup, jobs, pick, Rjob, sort, channels, things exec, wake, buttons, Hotkey, Hotjobs. Fully multitasking, allows many programs to run at once. Requires min of 256k expanded memory .... @£49.35b
Upgrade QRAM to QPAC II return master ..... @29.61b

PLEASE SEND SAE FOR A COPY OF THE QPAC II REVIEW

TONY TEBBY SOFTWARE (QJUMP)

QLFP (Micro/P disk interface upgrade) ..... @£15.51d
QTYP Type/Spell Checker ..... @£30.55d
QPAC 1-Desk top accessories, calander, alarm, calculator, typewriter, digital clock sysmon ..... @£22.56d

ZITASOFT SOFTWARE by Steve Jones

LOCKSMITHE copies M/DRIVE - M/DRIVE @£14.10c
4MATTER + LOCKSMITHE copies M/DRIVE - DISC .. .. @£23.50c
SIDEWINDER - High resolution printer driver prints full screens or parts of screens from postage stamp size to large banners. Prints sideways, invert, scale, mirror, text insertion ..... @£20.21c
SIDEWINDER PLUS - (for expanded QL's) includes all the features of above, PLUS multiple label printing, desk-top publishing files and printer driver for The LC10 JX80 24 pin and colour printers (Please state 3.5" disc or M/Drive) ..... @£23.97c
Upgrade to Sidewinder Plus ..... @£8.46c

QPOWER REGULATOR

Switch mode power supply, QL runs cool, stops, lock ups (due to voltage drop). Helps filter noise out. Simple to fit (no soldering) £25.85c

COLOURED PRINTER RIBBONS

Gold, Silver, Magenta, Orange, Purple, Brown, Green, Blue and Red.
Citizen 120D/Swift £7.99c
Cannon 1080A/1156A £9.87c
Epson FX80/LQ400/LQ800 ..... £5.17c
Epson FX100/LQ1000 ..... £7.99c
Panasonic KXP 1080 ..... £8.46c
Star LC10 ..... £7.52c

READY MADE LEADS

RGB QL TO PHONO £6.11c
RGB 8-6 PIN £7.52c
RGB 8-7 PIN HITACHI £7.52c
RGB 8-7 PIN FERGUSON £7.52c
RGB 8 PIN TO SCART £11.75c
6WAY PCC TO 25WAY 'D' £9.87c

24 Hour Order Line 0923 894064
OPEN
9am-5pm Mon-Thu
9am-4pm Fri

CARE ELECTRONICS
DEPT QL, 15 HOLLAND GDNS
GARSTON, WATFORD,
HERTS, WD2 6JN.
Tel: 0923-894064
Fax: 0923-672102

HOW TO ORDER:

By Post. Enclose your Cheque/PO made payable to CARE Electronics or use ACCESS/VISA. Allow 7 days for delivery

Please add carriage
a=£11.75 b=£3.76
c=No charge d=£2.35

PERFECTION
PERFECTION PLUS

NEW!

DIGITAL PRECISION LTD

Not just a word-processor - this one is THE word processor.
From the same inspired team who brought you classics like LIGHTNING SPECIAL EDITION and PC CONQUEROR, Digital Precision presents a product that will revolutionise the way you use your QL. Let us tell you how **PERFECTION** will do this.
Several hundred thousand QL users have grown familiar with the free word processor that was bundled with the QL. On the plus side, its use could be mastered in a few minutes thanks to its simple menu system, and it is reasonably WYSIWYG (What you see is what you get) in appearance. On the minus side, it is very slow, sometimes idiosyncratic (in what it prevents you from doing, or the roundabout way in which it forces you to go about things that should have been straightforward) and very many commands that we think should have been provided with it simply weren't. Valiant attempts to accelerate it by "patching" it have achieved only a 20% speedup.
There is, however, no getting away from the fact that the majority of QL owners still use Quill as their main program. They have grown used to the user interface of the bundled programs, and are reluctant to invest time in learning some totally incompatible system, whatever its claimed advantages might be. So - using their ingenuity, for QL people are an ingenious tribe - users have put up with the inadequacies and slowness, and enjoyed Quill's friendliness.
You are probably just such a QL user yourself....
Now here is a product CREATED JUST FOR YOU.
A word processor that you can master in just a couple of minutes.
A word processor whose user interface uses precisely those keys that you would expect it - from intuition or experience - to use.
A word processor that is menu-driven (multiple page menus) so there is absolutely no need for you to remember anything or ever refer to the manual: the menu is on the screen all the time. A word processor that is intuitively obvious to operate: even more obvious than was the bundled one. If you have become at all used to the F3 interface, you will love this enhancement! A word processor that is designed for Absolute Beginners and Advanced Users, for Complete Wallies and for Albert Einsteins alike.
A word processor that is delightful to use for letters and documents of but a few pages as well as for articles, journals, magazines, books, theses or manuscripts hundreds of pages long.
A word processor that can unleash the power of your printer, whatever its make, and squeeze the very best from it. A word processor with a very flexible user-configurable printer driver: one, however, that you should never have need to configure(!), as it works as shipped with Epson-compatibles and most non-compatibles, and if you have an esoteric printer, it can utilise your existing configured Quill printer data file, automatically, if you want it to! With this word processor you do not have to buy any extra printer drivers - you get everything you need right from the start.
A word processor with full on-screen indication of character mode - bold (i.e. emphasised) appears bold on screen, underline appears underlined, italics appear in italics, superscript and subscript appear superscripted and subscripted.... Other "special" type modes - dependent on the capabilities of your printer, like switching fonts, pitches, NLQ/draft mode, proportionality, double-strike, or anything else you choose - are indicated on screen by variations in ink/strip colour combinations, just as the most advanced PC word processors do. You can even make up your own "attributes" to be displayed on-screen in a particular ink/strip. Combinations of attributes are permitted - the display copes fine. Never before have things been so clear and simple....
That **PERFECTION** manages to do all this is remarkable. That it manages to do it at all fast would be amazing. But the truth is, in fact, much much better. **PERFECTION** is by far the fastest word processor for the QL, being DOZENS of times faster than Quill on many operations (a minimum of five times faster than it on everything), and - yes - many times the speed of our own beloved and excellent Editor, and far far ahead of all the others.
This may seem impossible to you. Two years ago, before we started work on **PERFECTION**, it would have seemed impossible to us as well! But a remarkable bit of software technology has enabled us to achieve this incredible acceleration. Of course you don't need to know or understand how we have accomplished all this in order to enjoy to the full the benefits of **PERFECTION** speed: if you want to know anyway, look at the technical section later on.
If you already use Quill or ANY OTHER QL WORD-PROCESSOR, you will be overjoyed to know that **PERFECTION** can load your existing saved files (_doc or _lis or ASCII) directly, with no conversion process required. This - together with the automatic reading of existing printer driver data - takes all the trauma out of the move to an exciting new system! And **PERFECTION** files are usable with PC/ST/Amiga word processors too.
PERFECTION comes with a multi-function configurator that allows you - if you want - to tailor-make a version specific to your tastes. Practically everything that is settable at run-time is also pre- configurable, making **PERFECTION** comfortable to operate. As you become more familiar with **PERFECTION** and no longer need the menu options to be visible all the time, you can toggle the menu off, freeing all the screen for your document. You can configure **PERFECTION** so that on startup the menu is either visible or not. As you become even more familiar with **PERFECTION**, you can opt to bypass the menu system entirely, and use alternative direct keypress commands to access **PERFECTION**'s power even more rapidly.
PERFECTION natively multitasks (of course) which means that without any other tools you can run multiple copies of it simultaneously, as well as run it at the same time as other pieces of software. Even if you choose to run only one copy of the program, you still have the option to look at more than one part of the document at the same time. You can take a "snapshot" of part of the document, and keep that snapshot in view as you edit a totally different area of the document. Ideal for indexing or cross-referencing. You can set up macros so that making a glossary is easy. Also, you can have any number of blocks - not just one - defined in the document. You can undo/edit attribute changes with a single keypress - there's no need to laboriously "paint" over areas or navigate to the start and end of a highlighted area in order to adjust the attribute!
Being able to cope with human error is an important part of **PERFECTION** philosophy. For example, not only is there an Undo option, but you can also ESCape from any command. When you have right justification on **PERFECTION** will add pseudo-spaces to pad out the line. Pseudo-spaces look like spaces and print like spaces but when you left justify they are removed while real spaces - the ones you have entered (via the SPACE bar or TAB) are not. This means if you accidentally right-justify tabular or columnar data, a simple left-justify will get it back to its exact original state. Most other word processors do not distinguish between spaces you have entered and spaces they have inserted, and hence cannot auto-recover.
Many users need the uuse (sic) of a spelling checker with their word processor. Adequate spelling checkers already exist for the QL, and for users who either do not want a spelling checker, or who do not want one as yet, or who already have one and are on a tight budget, we supply a version of **PERFECTION** without any built-in checker. But to get the best out of **PERFECTION**, we also supply it bundled with a dedicated Spelling checker of unsurpassed speed. There are even two levels of dictionary supplied (you get both) - the larger one is 225,000 words (no more hassle of having a checker which doesn't know the words you use: this dictionary is about 400% larger than its nearest competitor!) and a compact one: use the latter if you are short of memory, or when your document is really huge. You can add new words to the dictionary as well as create new dictionaries. With either dictionary **PERFECTION** PLUS one checks as you type, or checks saved files, or - BEST OF ALL - spell-checks interactively from any one point in the document to another....
If you already have our Editor Special Edition and use it for documents, database work or programming, you will find **PERFECTION** a wonderful treat. **PERFECTION**'s WYSIWYG behaviour, greatly enhanced document facilities ("tells you everything" status line, available word/line/character counts, regular and forced page breaks, headers and footers), menu-driven options and VERY MUCH GREATER SPEED make it an ideal upgrade. There are hundreds of detailed changes - to give but one: paragraphs do not need to have a blank line between them in order to distinguish them any more. There remains an area, however, where Editor Special Edition remains supreme - the editing of "non-printable" data, the ability to handle the entire ASCII character set from codes 0 to 255. So if you are a technical or semi-technical user and do not have either Editor Special Edition or **PERFECTION**, your best buy is the two programs together (they can interact, coexist, work simultaneously and have fully-compatible file formats). You will then get Editor Special Edition at HALF PRICE (Special Offer - limited duration).
The characteristics of a good database are its ability to Store, Retrieve and Manipulate information rapidly. By this criterion, this word processor makes an ideal database system too, as it is blindingly fast and flexible. Forward and backward 'Search' takes at most a couple of seconds, even when you have a document that fills an 896K Trumpcard system to the brim! Cursor navigation is also unbelievably fast and smooth, with an accelerating rate of scrolling if you indicate impatience. And there are macros, programmability and more for the more advanced user. If you have been unhappy with the speed or complexity or non-programmability of your existing database, **PERFECTION** will solve your problems.
PERFECTION can even access your existing Archive export files.
And if you want full desktop publishing capabilities (the use of fonts that your printer does not possess, and graphics) interlinking **PERFECTION** with Professional Publisher is a doddle. Use **PERFECTION** for creating, editing and manipulating, and "pour" the result into Pro Publisher.
But first and foremost **PERFECTION** is a user-friendly, familiar user-interface, stand-alone WYSIWYG dual-control (menus or direct commands) word processor of enormous power and blistering speed, which (for the first time) makes output to printers hassle-free. There is nothing else like it or even remotely as good as it on the QL or on anything else. **PERFECTION** is our best yet.
PERFECTION is for you whether you hate your existing word processor, are indifferent to it or love it. **PERFECTION** will let you forget about all the technology and concentrate only on the writing.
PERFECTION costs just £79.95 including integrated printer drivers, ancillary programs and jargon-free, friendly but to-the-point documentation (that you will probably never need to read through!). **PERFECTION** PLUS comprises **PERFECTION** plus the dedicated Spelling checker with dictionaries and costs just £119.95.

TECHNICAL INFORMATION ON PERFECTION

You don't actually need to read or understand this.
PERFECTION gets its superb speed from two sources. Firstly, **PERFECTION** - unlike virtually any other word processor - is written entirely in 100% hand-written machine code. This gives us a considerable speed advantage over compiled alternatives. Had we written **PERFECTION** in a high level language it would have been 4 times slower, 6 times bulkier and taken us a great deal less time to produce. You reap all the benefits of our hard work.
The other source is design. There are two formats for internal data storage for character handling programs. Many store data serially, in a long stream of characters. Ones like Editor store data as lines scattered through RAM, with a table of pointers to the lines - a far more advanced method. The first format has the advantage that it is cheap to program - the user pays the cost in terms of performance, with sluggish block-defining/moving, navigation and insertion. The second format has advantages including instant random access to any line and quick insertions and deletions - the disadvantages may include heap fragmentation that will result from repeated grabbing of small chunks of space (garbage collection may be required periodically if space is short). Both formats share the disadvantage that "global" changes made to a part of the document - say a switch to bold at the top - will take a long time to filter down through the system to become visible on-screen at lines at the bottom.
PERFECTION uses a variant of the second format that does not have its disadvantages. Data is stored in RAM in optimally-sized chunks - a chunk being roughly the size of several screens. Each chunk has a control information area within it about the number of lines etc within it, the display status at the start of it (say bold on, italics/underline etc off). Whenever you are editing, the relevant chunk(s) are instantly loaded into a large work area that has slack space at both top and bottom. That means that you can add or delete a great amount of data instantly, without **PERFECTION** having to bother about updating anything but the work area. Only when you move over the edge of the work area will **PERFECTION** need to housekeep outside the work area: the housekeeping itself is then very very fast, as only control information areas need to be updated. There is no need for a general scan through all following text. There are many more speed and power advantages to our system. There is one big disadvantage - it is an absolute nightmare to design and implement! Fortunately for you, you don't have to know anything about it - it just works like clockwork, automatically and behind the scenes.
Other elements of **PERFECTION** design to enhance performance include lazy screen (when you keep a key pressed in order to get somewhere, we stop updating the whole screen and instead just scroll the line your cursor is on) and lazy attributes (where in a huge document of hundreds of pages you do a long jump - say from near the top to near the bottom, in one go, and we have not yet resolved the attribute status (say underline on) of the area you want to get to, we don't hold up the display for even one hundredth of a second while we are computing attributes, but display the new area immediately without any pause - the attributes will 'catch up' a second later: you will only see this if your document is very very big and you navigate in huge leaps). Also, there is a garbage job running all the time in the background, doing whatever internal tidying up and optimising is needed when you are not doing anything (with **PERFECTION**'s speed, even if you are typing at 200 wpm the program is sitting twiddling its thumbs for 90% of the time as it awaits input!). Consequently, **PERFECTION**'s internal tables are always in a PERFECT state. Both lazy screen and cursor acceleration are user-configurable, incidentally.
There are dozens of other more localised ways in which **PERFECTION** performance is obtained. For example, **PERFECTION** has built-in knowledge of statistical distribution of occurrence frequencies for the various alphabetic characters in English and other European languages. It uses this data as follows: if you ask **PERFECTION** to search for the word 'praxis' in your document, we won't look for the 'p' first. Instead, we automatically look for an 'x' (less occurrences of 'x') and having found 'x' then resolve whether it is embedded within an occurrence of 'praxis' (if not, we search for another 'x'). Obvious? We thought so. But no one else appears to be using this excellent trick. Or dozens of other tricks that we'd prefer to keep to ourselves...
For those with advanced needs, **PERFECTION** features include full programmability - more than just macros - with the ability to save and re-execute programs. There are over a hundred commands. You will be relieved to note that **PERFECTION**'s file format is very clean, containing one short header (giving the margin/TAB etc data for that document) and then exactly what you typed in (no mass of pointers or counters). Changes of attribute (bold, NLQ, underline etc) that you have opted for are stored as control characters (we document the structure) in the appropriate places in the file - note that while the control characters themselves (as opposed to their effects, which are WYSIWIG'd on-screen) are invisible when you are viewing the file, you can edit/delete them (search for the next or previous bold text, say!) and even program the access to them (swop all bold for double-strike plus underline)... You can even opt to Export so the header is suppressed, to enable its direct use as a programming or technical front end, or to allow its output to be read in by other word processors (QL, PC or whatever).
The net result of all this is that in terms of features and performance, **PERFECTION** running on a QL will beat most word processors even running on state-of-the-art £7000+ 486 PCs... In a nutshell, **PERFECTION** will blow your socks off.

LIGHTNING SPECIAL EDITION
LIGHTNING

Until the autumn of 1989 the fastest way of speeding up your QL display was to buy **Lightning**, which greatly accelerated QL text printing, graphics and maths, without affecting compatibility at all. NOW you can buy **Lightning** Special Edition, which is significantly faster than **Lightning** and does a lot more! **Lightning** Special Edition is simplicity itself to use. Once it is loaded ALL programs will AUTOMATICALLY benefit from the enhancements it provides. If you are using a QL without **Lightning** you are probably a little pale (quote from John Norton of Sector Software), you should get out and about more... Go to some QL shows or meetings where you will see **Lightning** in action - or take our word for it. If you don't have **Lightning** you are WRONG.
Lightning Special Edition works by automatically (I know we keep using the word, but it is the only one that is really correct here) and instantly replacing QL ROM code (or Minerva code, for that matter - Minerva and **Lightning** complement each other superbly) that has usually been optimised for space, with extremely high speed routines written by us that do the same job but much faster. Screen output speed gets accelerated by factors from over 1.5x to over 10x (about 2x-4x is representative), graphics are drawn twice as fast (points are plotted 5 times faster) and internal maths is speeded up by 2x-5x (you can even vary the precision). There is virtually no cost in RAM (for example, you can still run Quill with a fairly large document on an unexpanded QL with **Lightning** Special Edition).
The Special Edition is supplied on EPROM plus disk/cartridge: if you already have something precious plugged into the QL's EPROM socket (at the rear), there is no problem - all the EPROM's functionality is duplicated on the other medium!
Lightning Special Edition provides more than acceleration - you can dynamically adjust channel parameters - like ink, paper, font, screen position, use over 80 fonts, a null device, a character drain and all sorts of other interesting gadgets.
Lightning Special Edition installation has been totally automated, and will not present you with complications no matter how computer-naive you are.
If you cannot afford the Special Edition, get **Lightning**. Refer to its review in September 1988 QL World to see how effectively **Lightning** acquitted itself.
Both of these programs transform the QL into an altogether more zippy, business-like, efficient, enjoyable machine.

PC CONQUEROR WITH DR-DOS v5.0
PC CONQUEROR

Terrific though we know the QL to be, we do feel the pressure to be "PC compatible" in today's world. There is increasing demand to be able to bring home and run the programs we use at work (or the other way around!), and to have access to the vast storehouse of PC software: word processors, databases, spreadsheets, expert systems, accounts and financial modelling packages, vertical market applications, visualisation aids, graphics/CAD/PCB designers, languages/compilers, operating systems, environments, utilities, adventures - you name it, there are scores of each type readily available for the PC. And thousands of shareware/PD programs too, most for the cost of a blank disk plus postage.
If you buy PC **Conqueror**, you will be able to run these programs! To boot up PC **Conqueror** takes 10 seconds from the F1/F2 prompt: thereafter, your QL is a HIGHLY compatible PC clone (indeed, more compatible than some "real" PCs). **Conqueror** is all-software. There is no comparison in quality between **Conqueror** and its predecessor: **Conqueror** has ALL the features of Solution (read the details later in this ad if you are unfamiliar with Solution's legion facilities), but is almost TWICE as fast: this has come about by our careful rewriting and optimising of Solution's code. As if the colossal speedup was not "enough", **Conqueror** (unlike Solution) runs perfectly even with PC software that makes various "non-legal" calls to the PC operating system. **Conqueror** runs with virtually anything that will run on a PC: QL Worlds from December 1989 to March 1990 listed several hundred PC programs/utilities found to work with **Conqueror**. It is simpler to say that we have yet to find a program that runs fine on a standard PC that doesn't run with **Conqueror**: we are aware, however, of programs that will run with **Conqueror** but won't run on some PCs!
Because in **Conqueror** we've cracked the problem of detecting when the PC screen has been changed, we need not slavishly update the screen many times a second (taking precious time away from the main PC-emulation job) as did Solution. Instead, we update the screen instantly it needs to be updated. This simple to understand but very hard to implement modification gives **Conqueror** additional (over and above what we've already discussed), "tunable" acceleration, as well as absolutely smooth echoing of keyboard input to screen (Solution could be a bit jerky when you typed quickly).
Conqueror's new features include a more flexible configurator and a better diagnostic and supervisor option, an enlarged manual (**Conqueror** itself is more compact!) with a troubleshooting chart, and a new mode of operation (in addition to the "normal" one of reading/writing PC disks directly) which allows you to create mini PC environments - you select the size, location and name - on any QL device (including floppy, ramdisk, hard disk and even mdv) which look like files from QDOS (and can therefore be copied with SuperBASIC's COPY!) but are indistinguishable from PC drives from within DOS (**Conqueror** works with all versions of DOS).
If you do not have legal access to a copy of DOS, you need to buy DOS too (DR-DOS or MS-DOS)- but we sell the complete DR-DOS (with Viewmax, Shell, Cache and all system utilities) at c 1/2 price!
Of course QLs are better than PCs - but QLs that are PCs as well are better still. We will leave the last word to people who have already bought **Conqueror**. All these sentiments are unsolicited.
*"I wish to congratulate you on the excellent work you have done on **Conqueror**. The improvements in performance over Solution are astounding. Well done!" B.C.Papegaaij, Netherlands. "I am highly delighted with this new emulator. (Apart from the speed-up) it also appears to be more tolerant." L.Chandler, Peterborough. "Congratulations on bringing such a fast PC emulator into the world - on it, even Wordperfect runs at a reasonable speed." R.Williams, London. "I'm impressed with the improvement in speed over Solution." P.Vervoort, Netherlands. "Thank you for your prompt service. I have **Conqueror** up and running, and congratulate you on an excellent piece of software." G.Leagas, Hartlepool. "On some benchmarks almost as fast as a PC." P.Johnson, Stoke on Trent. "**Conqueror** is still a whole lot faster (even) without Lightning than Solution is with the assistance of Lightning." P.Christie, Glasgow, who went on to praise **Conqueror** for running software Solution couldn't handle. "**Conqueror**, to which I upgraded from Solution, is a delight to use by comparison!" B.Gouldwell, Dunipace..... V.Pakanen, Finland sums it all up rather well with - simply - "Excellent."*

PROFESSIONAL PUBLISHER

To show you a little of what our **Professional Publisher** can do, we have prepared our last advertisement using it. Notice from our May 1990 advertisement how we can wrap the result around graphics or in fact anything, of any shape. When we wrote **Professional Publisher** (PP), we knew it was a very special sort of program. PP can produce pages of quality - virtually indistinguishable from those prepared on professional typesetting kit, the only limiting factor might be your printer: however, while the very best output output from PP will be obtained from 24 pin models and lasers, you will be stunned by what PP can squeeze out of the humblest 9-pin machine. Great care was taken in the design of PP so we were absolutely sure that no actual knowledge of, or practice with, desktop publishers was required in order to use it (the '**Professional**' in '**Professional Publisher**' refers to the output quality, not the level of operating skill required. When you use PP, you will notice that at each and every stage a menu is available (there are getting on for a hundred menus in total) with a list of options selected by using either the cursor keys

There is context sensitive, on-screen help too. When you get more experienced with the program, you may select Command mode (using the Enter key) and choose operations directly, bypassing the menu system. PP is more user-friendly than any page-making program we have ever seen on any computer, period.
Let us talk you through how you might choose to produce a page or succession of pages. This is just one way you might proceed: PP does not impose any sequence of steps upon you, and you can omit certain operations altogether. You will have pre-configured PP to boot up with a generous lot of fonts (you select which ones you are likely to want - of course you can load in additional ones, or discard existing ones, at run-time too). You could then set the required page dimensions and orientation, as well as not-necessarily-symmetric margin, grid, gutter, column and navigation-guide positions (yes, half the PP manual is a glossary) - you could have pre-configured PP for these too, or loaded in alternative layouts (layouts are distinct from page contents) you've created in previous sessions. If you don't set layout we'll use the default, or the one used for the previous page. Now you would plan the page in detail. Laying out graphics (if any) comes next - you can create these in PP itself, with its superb rubber-banding, dozens of brushes, palettes, texture-fills and so on. Alternatively, you can load in screens created elsewhere, including Eye-Q, Easel, any other graphics programs or digitiser, into a cut and paste buffer where a dozen tricks (including resizing, slanting, scrolling and texturing) are available, and then take the finished product onto the page. This done, you might insert headlines or captions, selecting from the dozens of fonts available. Each font can be manipulated in billions of ways (yes, we mean thousands of millions): to give but two examples, you have a choice of 32 slopes for italics for the font, and dozens of aspect ratios are selectable. Now you might opt to get the main body or bodies of text down on the page. As fonts are defined to great accuracy (upto 2304 pixels PER CHARACTER!) jaggedness is a thing of the past, and visually the choice of fonts can only be described as stunning! You can do this either by directly typing it into cursor-dragged boxes (with all the options you would expect from a dtp system, and a few more besides), or by loading it in from file created by Quill, **PERFECTION**, Editor or other word-processor. The latter method is better (because you retain the text as a character stream rather than as pixels when you save the file). Highlights such as bold, underline etc which you may have inserted into the text are preserved. Indeed, you can control PP's operation from within the text file itself. If you are an advanced user, you can even teach PP your own mnemonics, so that it switches between different styles and modes as it encounters instructions you put into your text file when you created it! The imported text file is editable within PP. It is up to you to decide where the text is to lie - PP places no restrictions on either the number or the shape of the windows into which the text is to flow: they need not be rectangular, and can have any irregular border, and can even overlap or be contained one inside another! You can freehand-draw (there's excellent rubber-banding to help you) the window borders as you choose, to get any effect you desire, to fill any space you wish and to avoid any existing material already on the page (or to reserve room for new material). Amazingly, within the window the text will all be perfectly micro-justified in the font(s) of your choice, however bent or contorted you made the border. Text will flow automatically from one window to the next either until you have run out of text or out of windows. There are many text formatting facilities: you can select word-wrapped, force-broken or hyphenated, and you can specify minimum numbers of "pre-hyphen" and "post-hyphen" characters so that absurd hyphenations are avoided (if no sensible hyphenation position can be found the word is wrapped instead). There are so many fine-tuning controls here that the rest of this ad could be devoted to describing them and would still leave things out! We will have to content ourselves with but one example: with micro-justification (pixel by pixel spacing, not crude character by character stuff) we even allow you to specify what % of padding space is to be allocated between characters and how much between words! Text work completed, you can then put in the final touches by adding borders, shadows, patterns or designs, overwriting or slipping under or combining these with existing material, repositioning parts of the page if necessary. The end result - be it for a letter, letterhead, document, manual, article, newsletter, magazine, book, thesis, ad - is far better than you have any right to expect from a piece of software costing under £2,500, let alone under £100...

PROFESSIONAL PUBLISHER TOOLBOX

NEW!

For Professional Publisher users - this useful addition not only supplies several man years worth of beautiful high definition fonts - including familiar types like Roman and Universal - but also contains many smaller fonts, more clipart and programs to load Sector Software clipart, filter text before importing into Professional Publisher, save parts of Professional Publisher pages as screens (for importing into any graphic program - like Eye-Q - or manipulating via SuperBASIC) etc. Excellent value.

FONT ENLARGER

For Professional Publisher users - loads of large fonts are automatically created by this multitasking utility, as and when you need them (or in advance), by enlarging existing smaller smaller fonts from PP itself and from Lightning Special Edition and hordes of other sources: with this there is NO jaggedness at all. A font editor for small and large (hdf) fonts is included.

GRAFIX

Scaleable output for all our desktop publishers on 9- and 24- pin printers: a useful alternative to the built-in drivers.

EYE-Q

There is no way to describe **Eye-Q** except as the best graphics program for the QL. This master is now four years old, and we have never felt the need to change anything. Its use is characterised by absolute simplicity, speed and power - it has that indefinable precision "feel" that is just right. All the expected manipulations are provided. Whether your needs are technical drawing, labelling, design, illustration, freehand work, copying - or just having fun, **Eye-Q** will not disappoint. Of course it is menu driven with context-sensitive help. The system takes 5 minutes to learn. The variable zoom and fill facilities, anti-fingerslip measures, cursor acceleration and so on make **Eye-Q** a classic in its own time.

ULTRAPRINT

To get the best printer output from Eye-Q or any other graphics program from any other source, **Ultraprint** delivers. An amazing 22 styles to choose from: enhance contrast (for line output) or gradation (for pictures) and vary magnification... A printer without **Ultraprint** is no printer at all.

MEDIA MANAGER SPECIAL EDITION
MEDIA MANAGER

MMSE is a joy to use. Whether something has gone wrong with a disk or tape ("Not found", "Not a valid Quill file", "Bad or changed medium", "Read/write failed" etc) or whether you want better control over your programs and data, **MMSE** should be to hand. Virtually any calamity can be recovered from automatically: all permutations (accidental deletion or part-overwriting,

directory but OK map, bad map and directory, OK map and directory but bad file sectors, unknown fault, power glitch corruption and so on) have been carefully thought through and catered for. If nothing is wrong, but you just want to explore and understand more about your system, you can potter to your heart's content, assisted by the clear and packed-with-facts manual. Dozens of different diagnostic printouts can be produced. The whole system is menu-driven, with context-sensitive, on-screen help for every option. The speedy Sector Editor is a positive delight: the collector file facilities, bulk recovery, auto-navigation, skipping through the medium in physical, file (if map), logical (if no map) or uncollected/logical (if destroyed map, and because of "chequered" history with lots of overwriting/deletions no one-step recovery available) sequences must all be experienced to be believed. **MMSE** is extremely simple to operate, and assumes no advance knowledge whatsoever.
Alternatively, if you wish to tidy up your disks or cartridges, **MMSE** allows you to change volume format names, sort directories into alphabetic, date or size order, analyse file contents and histories, change case of filenames, move data/programs to/from alien-format disks, introduce or break copy-protection systems (illegal use prohibited!), **MMSE** can and will deliver the goods. It is absolutely superb.
The standard **Media Manager** is much less powerful, and less easy to use. It is only for those on a tight budget.

TOOLKIT III WITH ROM
TOOLKIT III

Virtually everyone with a disk system has Tony Tebby's fine TK2 Supertoolkit on board (usually built into the disk interface). **Toolkit III** - which works whether or not you have TK2 - takes off where TK2 ended, adding about 70 new commands and enhancing many existing QL and TK2 commands. **TK3** is for everyone with a QL. You can get this system on cartridge/disk, with or without a plug in ROM cartridge in addition. The documentation is complete and very comprehensive. Some of the added commands are:
ADIM * ADIMN * AND_L * ATYP * BASREF * BV_BASE * CHANNELS * CH_BASE * CINT * CLOSE% * DEVLINK * DIR_USE * DITS * DIV_L * EOR_L * EXTRAS * FACC * FLP_SEC * FLP_START * FLP_TRACK * FLP_USE * FRAC * ISFLT * ISINT * KEYS * LARRAY * LOWERS * MEMCOPY * MEMSWAP * MJOB * MJOB_W * MOD_L * NFS_USE * ODD * OM_INIT * ONPIPE * OR_L * PEEK_F * PEEKS * PEND * PIPE_ * POKE * POKE_F * PRED * QDOS$ * QIN * QOUT * QTEST * QWAIT * RAM_USE * REPLACE * REPLACES * RESET * RJOB_A * ROUND * SARRAY * SEARCH * SETDIR * SETDIR_A * SETHOST * SETNET * SETRO * SETRW * SETSYS * SETUSER * SGN * SORT * SORT_I * SUCC * TK3_EXT * UPPER$ * USER * WN_BASE * WSETHOST * WSETNET * WSETRO * WSETRW * WSETSYS * WSETUSER
Whether or not you can program, **Toolkit III** is of great use!

QFLICK CARD INDEX SYSTEM

Few users actually require all the facilities of a complicated database like Archive. **QFlick** presents a very convenient alternative - a very fast, simple to use card-file database, with easy to learn, snappy search and navigate commands and clean file-handling. You can move Archive data to/from **QFlick**. You can run multiple copies of **QFlick**. And **QFlick**'s data is organised so it is easy to program from SuperBASIC, even for tyros!

PERFECT POINTER TOOLS

This excellent program gives you an on-screen pointer (arrow) environment of incredible smoothness, and 6 utilities with it. To explore the world of QPtr, Things, Hotkeys, Window Manager....

QKICK MULTITASKING SYSTEM

A pull-down menu controlled multi-tasking front-end, ideal for running in the background and giving you notepads, file-handlers, quick backup, clock, diary, mini-database, calculator etc etc.

DISKTOOL WITH QUICKDISK

An exciting way to accelerate disk access by upto 30%, add password protection to disks and to optionally increase disk storage capacity by 36K to 1512 sectors! All this works while maintaining full compatibility and normal disk control...

DIGITAL C SPECIAL EDITION
DIGITAL C COMPILER

Superb C compilers these - fast in execution, they produce extremely speedy and concise code. No-nonsense documentation is included. The Special Edition has many more features, including pointers, long pointers, structures, >64K code sizes, direct access to traps and vectored utilities, and is twice as fast because of its more efficient C/QDOS libraries.

TURBO BASIC COMPILER + TOOLKIT

This state of the art system will automatically convert ordinary SuperBASIC programs - the sort you buy, write yourself or type-in from magazines - into machine code, the language of the 68008 CPU, the brain of the QL. Such pure machine code programs run "directly", without the need to be interpreted by any intermediary system. This direct execution makes them MUCH faster in execution than BASIC. **Turbo** also adds a host of useful high-speed commands (called "toolkit extensions" if you are fond of jargon). Here are some timings, all carried out on a JS Trumpcard QL, to give you a taste of just how much **Turbo** can improve things:

	Iterations	SuperBASIC	Turbo'd	Speedup
Empty FOR...END FOR Loop	30000	49 sec	1.3 sec	38x
Empty REPEAT Integer Loop	30000	151 sec	2.4 sec	63x
String concatenation	3000	448 sec	0.4 sec	110x
Search through memory	300000	1410 sec	1.5 sec	900x

Turbo's automatic conversion process, called compilation, is as simple as this: (1) Boot up with the **Turbo** disk (2) Load in or type in your BASIC program (3) Enter the word CHARGE, and watch the friendly front-end menu pop into view (4) Choose a filename for the machine code task that is to be generated and (5) Press the SPACE bar. **Turbo** does the rest! Compilation is a one-off process, and is very fast too - it takes little more time than LOADing the original program did! Once compilation is finished, you have a machine code version of the original program. Start this with EXEC, just as you used to invoke the original program with LRUN: besides the tremendous difference in running speed, you will notice that the program loading time is cut down to a few seconds at most (big SuperBASIC programs can take half an hour or more to load). The EXEC mechanism also allows you to multitask programs, something impossible with SuperBASIC, as well as manipulate their time-priorities, link them together, exchange data and even share parts of their code while executing.
If you are an advanced user, **Turbo**'s numerous fine-tuning facilities, 200-command toolkit (a terrific complement to the famous Supertoolkit) and 300+ page manual will be irresistable.
If you are a beginner, you will wonder how you ever did without **Turbo**'s program diagnoses and auto-correction.

SOLUTION WITH DR-DOS v5.0
SOLUTION

This program transforms your QL into a pretty compatible - albeit not fast - PC clone. **Solution** will run over 95% of the "big name" PC software you have read about, missing out only on programs that make illegal use of the PC's operating system. **Solution** works solely from software so you don't have to worry about ripping your QL to pieces to fit anything, or have anything hanging out of the back. Just boot up the **Solution** disk and you will be using a PC, which will then ask for a copy of DOS (any) (just as it would if you were using a "real" PC). End of story - you are now using a PC. There are very few restrictions: both mono and colour CGA graphics are supported. 479K is available for PC software on a 640K machine and 667K when using Trumpcard - more than you will get on a PC or XT! Speed can be increased by using Lightning Special Edition but in final analysis just can't compare with Conqueror's speed). Because your newly aquired PC is really a QL you can multitask two or three PC programs (try doing that on a "real" PC!). You can also run QL programs alongside PC programs (DON'T try that on a "real" PC!). Converting files (data in either direction) between QL and DOS is no problem and you can re-configure the QL keyboard if you wish.

PROFESSIONAL ASTROLOGER
PROFESSIONAL ASTRONOMER

Our use of the term "Professional" in the name of an application program does mean that the quality achieved will meet or surpass the highest professional standards for that application. The term does NOT mean that you have to have the knowledge of a professional in order to get the best out of the programs.
Astrologer teaches you astrology from scratch, and enables you to produce reams (if you are short of paper, you can choose exactly how much) of narrative printout giving a person's horoscope, personality delineation, year-to-year life overview, detailed day-to-day (in fact, minute-to-minute!) predictions, as well as two-person compatibility interpretations. Also provides all the technical readouts, charts and zodiacal wheels you would expect. It is extraordinarily fast (there is a great deal of very clever maths within it) and it performs the whole computation in under a second. The author of the manual is the author of this advert, so you can expect a lucid and humorous read! Whether or not you believe in astrology - indeed, especially if you do not - this program is one that you cannot afford not to have. Scores of detailed readouts for famous people are supplied, incidentally - very interesting reading they make too... Discover Mrs Thatcher's secret yearnings, explore yourself, play the Stock Exchange...
Astronomer is an extremely efficient solar system computer, with planetarium views, planet faces (with shadows/eclipses), five different co-ordinate systems, 1sec=1day cinerama, etc.
Astrologer + Astronomer is supplied at a very low combined price.

ACT SPECIAL EDITION

The **Adventure Creation Tool** is for every programmer or putative programmer. Whether or not you have any interest in adventures, you will find something useful here. Animated graphics, data compression, language design and parsing, maps, object-oriented control and much more, with an excellent educational manual too.

3-D PRECISION CAD SYSTEM

2-D and 3-D design and manipulation, at a speed sufficient to permit real-time animation! Whether or not your interest is serious, 3DP will change the way you look at the world around us. The variation of viewpoint, perspective and magnification is very smoothIn addition to dot-matrix output, plotters are catered for.

SUCCESS

Run CP/M programs on your QL! What more is there to say, other than that after the PC family, no more common system exists than CP/M, with thousand of cheap programs... And **Success** is fast!

THE EDITOR SPECIAL EDITION
THE EDITOR

If your needs are for a technical Editor, or for full access to the entire ASCII character set (to handle machine code or compressed data files), or if your budget cannot stretch to **PERFECTION**, then this is the program for you. Editor is command-line driven and programmable. The Special Edition version is certainly better than the standard version: that is because the standard one contains only as many features as we could get to fit into an unexpanded QL. Both are fast and flexible, and very powerful indeed in the hands of the intelligent. Not a word processor, Editor's a way of life.

SPECIAL DESKTOP PUBLISHER
DESKTOP PUBLISHER

Both these WYSIWYG ("What You See Is What You Get" dtp systems are excellent in their own rights - it is only when you compare them with the stunning Professional Publisher that you become aware of their shortcomings. You won't get fonts as large or smooth as with PP, or wrap-around graphics, or as sophisticated a printer driver or text/graphics file import facility. You will get a very workmanlike tool, capable of producing output that the computer press described as fantastic and superb.... The standard edition is the ideal if you do not have a disk drive: if you do have one, go for the Special version, which correspondingly has more features including textures, large windows, better drawing and improved command entry. All upgrades are possible, and there is only a £10 penalty for doing it in two stages. So if you simply cannot afford PP, one of this pair is certainly for you.

SUPERFORTH COMPILER WITH REVERSI

Why not learn FORTH, the most logical computer language of all? This superb FORTH-83 compiler produces stand-alone multi-tasking code of speed comparable to C. **SUPERFORTH** source is even portable to other machines! The manual teaches you the language.

IDIS SPECIAL EDITION
IDIS

Machine code (from other people's programs, toolkits and the ROM) is unintelligible until you put it through IDIS, the intelligent disassembler. IDIS Special Edition automates everything it possibly can, and requires no human intervention. It even sorts out subroutines, replaces addresses with names, untangles data

MICROBRIDGE

Never be short of a four for Bridge again. Superb bidding tutor included, based on random hands dealt with lightning speed. Manual a masterpiece. Understands and obeys ACOL and much more.

SUPERCHARGE SPECIAL EDITION

If you have an unexpanded QL, or cannot afford Turbo, but want SuperBASIC programs to go faster, **Supercharge** is the answer. It has about half the speed of its big brother, is not as tolerant of badly-written programs, and lacks many of Turbo's features (like linking, program sizes >64K etc); nonetheless, it is the compiler about which we recieved over ONE HUNDRED happy letters from satisified users all using the word "Excellent" to describe it - and hundreds more who used other equally complimentary terms. The only gripe was about the Lenslok copy-protection, long since removed by us. So now **Supercharge** is wonderful....

SUPER SPRITE GENERATOR

SSG moves things about the screen rapidly, at machine code speed, directly from simple SuperBASIC. Any number of sprites (each with upto 16 frames for smooth realistic motion), 256 speeds, 256 planes, collision detection and dozens of special effects.

SUPER ASTROLOGER

A cut-down version of the Professional Astrologer - smaller horoscopes and manual, no interpretations for forecasting or compatibility testing. A marvellous buy at the price!

BETTER BASIC EXPERT SYSTEM

SuperBASIC is a super BASIC. If you want to improve your programs automatically, and learn as you do this, get **Better Basic**.

TRANSFER UTILITY

Copies files between devices, performing translates as it goes. Needs a ramdisk to run. Can move your microdrive material onto disk, so programs run from disk but you still have access to microdrives.

MONITOR

Check dynamic operation of programs - IDIS's ideal companion.

TERMS AND CONDITIONS)

* Our non-game programs are very comprehensively documented with A4 manuals averaging about seventy pages in length (the largest is 325+). They are 4-hole punched for easy binding and storage.
* UK purchasers - the quoted figures are all-inclusive. For the rest of Europe, add 5% (rest of the world, 10%) to the quoted figures to arrive at the VAT-free total (exports are zero-rated for UK VAT), inclusive of all freight and documentation charges.
* Acceptable forms of payment are sterling cheque drawn on a UK branch of a bank or building society, sterling postal order, Eurocheque made out in sterling, international money order in sterling, VISA/ACCESS/EUROCARD/MASTERCARD (specify expiry date), foreign currency cheque (add 10% conversion charge), cash, direct funds transfer (notify us in writing, and ensure that all charges are paid your end, or add 5%) to A/C 50327808 DIGITAL PRECISION LTD at Barclays Bank PLC (Branch code 20-79-44), South Chingford Branch, 260-262 Chingford Mount Rd, London E4 8JN.
* To upgrade from one version of a program to a superior program, send us the cartridge/disk. Except in the case of upgrades between program versions both with the same name, send the manual too. The cost of an upgrade is £10 plus the difference in current advertised price between the two programs. So the upgrade from DIGITAL C to SPECIAL EDITION DIGITAL C costs £30, and you would need to return the old manual as the program names are different. PERFECTION is not an upgrade to EDITOR, but SE EDITOR owners are allowed to claim a £5 reduction when ordering PERFECTION.
* Our programs are all user-transferable between cartridge and disk, are all free from ALL copy protection, and all work with all drives, toolkits, RAM add-ons and disk interfaces (except for early MCS interfaces, to which the emulators and media managers object). Users of the Microperipherals interface are recommended to buy the QFLP ROM upgrade from Care. ST/QL Emulator owners will benefit from a c2.7x speed increase on our software.
* All trademarks used or referred to in this ad are acknowledged.
* If you want Eye-Q, but want to use it with a Gigamouse, with QRAM or on a THOR, specify your intended use with your order.
* We supply unused cartridges @£3.50 each, £30/10 or £55/20.
* Digital Precision supplies quality hardware too - write/phone.
* Digital Precision (abbreviated by our admirers to DP!) is a trading name of DIGITAL PRECISION LIMITED, Company Registration No. 1833989, registered in England & Wales, VAT Reg. 420 1560 08.

PRICE LIST

	£	
PC CONQUEROR WITH DR-DOS V5.0	169.95	eT
PERFECTION PLUS WITH SPELL CHECKER	119.95	dT
THE SOLUTION WITH DR-DOS V5.0	119.95	eT
PC CONQUEROR	89.95	eT
PROFESSIONAL PUBLISHER	89.95	cT
PERFECTION	79.95	dT
TURBO BASIC COMPILER + TOOLKIT	79.95	aT
PROFESSIONAL ASTROLOGER WITH ASTRONOMER	69.95	aT
PROFESSIONAL ASTROLOGER	59.95	aT
LIGHTNING SPECIAL EDITION	49.95	+aT
DIGITAL C SPECIAL EDITION	49.95	aT
MEDIA MANAGER SPECIAL EDITION	49.95	dT
3-D PRECISION CAD SYSTEM	49.95	dT
SUCCESS	49.95	bT
THE EDITOR SPECIAL EDITION	49.95	dT
ACT SPECIAL EDITION	49.95	eT
TOOLKIT III WITH ROM	44.95	+a
IDIS SPECIAL EDITION	39.95	dT
EYE-Q	39.95	aT
THE SOLUTION	39.95	eT
SPECIAL DESKTOP PUBLISHER	39.95	ct
SUPERFORTH COMPILER WITH REVERSI	39.95	aT
MICROBRIDGE	34.95	d
PROFESSIONAL PUBLISHER TOOLBOX	29.95	eT
TOOLKIT III	29.95	a
QFLICK CARD INDEX SYSTEM	29.95	a
PERFECT POINTER TOOLS	29.95	a
SUPER SPRITE GENERATOR	29.95	a
SUPERCHARGE SPECIAL EDITION	29.95	a
PROFESSIONAL ASTRONOMER	29.95	fT
THE EDITOR	29.95	at
DIGITAL C COMPILER	29.95	at
GAMES COMPENDIUM (ALL FIVE GAMES)	29.95	a
QKICK MULTITASKING SYSTEM	24.95	a
IDIS INTELLIGENT DISASSEMBLER	24.95	at
LIGHTNING	24.95	aT
SUPER ASTROLOGER	24.95	f
DESKTOP PUBLISHER	24.95	d
MEDIA MANAGER	24.95	a
BETTER BASIC EXPERT SYSTEM	24.95	aT
FONT ENLARGER	19.95	eT
DISKTOOL WITH QUICKDISK	19.95	b
ULTRAPRINT	19.95	aT
MONITOR	19.95	aT
TRANSFER UTILITY	9.95	b
GRAFIX	9.95	eT
SUPER BACKGAMMON GAME	9.95	f
REVERSI GAME	9.95	f
ARCADIA GAME	9.95	f
BLOCKLANDS GAME	9.95	f
DROIDZONE GAME	9.95	f

KEY

! As well as a cartridge or disk, you get a ROM	+
! Available either on cartridge or disk	a
! Available only on disk	b
! Minimum 512K RAM:only available on disk	c
! Minimum 256K RAM:either cartridge or disk	d
! Minimum 256K RAM:only available on disk	e
! Available only on cartridge	f
! Compatible with THORs including the THOR XVI	T
! Compatible with all THORs except the THOR XVI	t

LIMITED PERIOD: SPECIAL DEALS ON OUR PROGRAMS

****

* Buy ANY *TWO* programs, get 10% off the total price!
* Buy ANY *THREE* programs, get 15% off the total price!
* Buy ANY *FOUR* programs, get 20% off the total price!
* Or buy *FIVE* or more to get 25% off the total price!

****

UPGRADES COUNT AS PROGRAMS FOR COMPUTING SPECIAL DEALS
DR-DOS (NOT BEING OUR PROGRAM) MUST NOT BE COUNTED

USER VERDICT

	Poor	Moderate	Good	Excellent
Installation				▲
Ease of use				▲
MS-DOS compatibility				▲
Use of PC's features				▲
Functionality				▲
Documentation				▲
Tutorial			Yes	
Value for money				▲

■ PC USER VERDICT This is the first time I've given any product an all 'excellent' verdict, which speaks for itself. ● PRODUCT DR-DOS 5

DR DOS 5.0

OPERATING SYSTEM
for PCs and
PC Conqueror

Digital Research
WE MAKE COMPUTERS WORK

*We can now supply the amazing new **DR-DOS v5.0**, normally sold for £119.95+VAT+P&P = £144, for **£80 all inclusive** (when bought with Conqueror/Solution,or by MSDOS users (who can at the same time send in the old Conq/Sol disk for a free upgrade) - else £90)! This best DOS has 600+ page manuals ViewMAX,Cache, online Help,screen Editor, command-line History, passwords, FileLINK +MemoryMAX, Ramdisk, 71 DOS files(1.4Mb), top compatibility & DR technical support. Our v5.0 is the **22/1/1991 NEWEST**, updated release. The PC press have raved about it and PC Plus (see left) gave it seven (7!) excellent grades. So get **DR-DOS v5.0 NOW!**

To: DIGITAL PRECISION LTD
222 THE AVENUE, LONDON E4 9SE

ORDER COUPON

Or telephone
081-527 5493

Please send me: _____

Name: _____ Address: _____

_____ Postcode: _____

Enclosed: CHEQUE/PO/CASH/VISA/MASTERCARD/ACCESS/EUROCARD for £_____ Signature: _____

Card No: _____ Expiry: _____

I use: (QL/THOR) (MDV/. DISK/5.25" DISK) (360/720/1440 SECTORS)

DIGITAL PRECISION LTD

☆ sinclair QL ☆

EEC LTD MAIN SUPPLIER OF Sinclair QL COMPUTERS & PRINTERS QLs FROM £125 PRINTERS FROM £130 inc VAT

The expandable system for small businesses, beginners, and experts.

QLs COMPLETE JM ROM FULLY TESTED AND WITH 6 MONTHS WARRANTY

Package includes PSU TV lead, QL Software 2.35, Quill – Word Processor, Abacus – Spread Sheet, Archive – Records, Easel – Business Graphics **£125**

Customers buying one of the QL Computers are given one year's **free membership to QUANTA** (Help, Newsletters, & 400+ Library of Programs – mostly free).
QUANTA MEMBERS can obtain a £5 discount
As above but fitted with JS ROM **£150**

★ £50 PART EXCHANGE ★

OFFER ON QL

Back up QL with JM rom £75. JS rom £99 consisting of keyboard unit, original Psion software wallet, leads and feet IN EXCHANGE FOR YOUR OLD QL
(if you want to keep it add £10 to the above prices)
OR
Spectrum Plus or 128K (working or not working as long as repairable). Power supply for QL if required £12.

PC/QL KEYBOARD AND INTERFACES

PC AT 102 KEYBOARD AND CONNECTOR	£25
QL TO PC INTERFACE FOR ABOVE	£75
CASE AND LEAD FOR EXTERNAL MOUNTING-TO ENABLE BOTH KEYBOARDS TO BE USED AND EASIER MOUNTING	£17

★ OUR PRICES NOW INCLUDE 17.5% VAT AND HAVE NOT BEEN INCREASED ★

QL Psion Software 2.35	pack £20	Quill, Abacus, Archive, Easel each	£12
QL Power Supply Unit	£12	Joysticks with QL lead	£12
Centronics Printer Interface	£27	Keyboard Membrane	£9.50
Complete User Guide	£20	QL ICs and Spares	ASK
Cartridges, Wallets of 4, new	£12	4 unused prog cartridges	£9

20 new cartridges in plastic storage box **£55**

Send S.A.E. for complete Price List, Softwear Leaflets, etc.

SOFTWARE QL Decision Maker £18. Home Finance £18. QL Games £9. Q-Liberator Super Basic Compiler (disc) £60. Q-Liberator Budget Version for (MDV) £30. Q-Load Superbasic Quickloader with Q-ref Programming utility (Mdv) £15. File Transfer Programs. QL to other systems including PC: Text Tidy 315. Discover £20 Multi-Discover £30
BUSINESS SOFTWARE (Suitable for standard QLs) Small Traders Pack £24.95. Invoicer £19.95. General Ledger £19.95. Set of three £50. Mail Merger £14.95. Filer Pack £14.95. Also Stock Accounting £39.95. (For expanded QLs). Send SAE for software leaflets, or phone

All prices include VAT. Terms CWO, Access or Visa. Minimum order £10.
Carriage £8.00 for printers, monitors & QL (overseas £20.00) other items £3.00 (overseas £6.00)

EEC LTD

18-21 MISBOURNE HOUSE, CHILTERN HILL, CHALFONT ST PETER, BUCKS. SL9 9UE
Tel: 0753 888866 Fax: 0753 887149

JOCHEN MERZ SOFTWARE

Im stillen Winkel 12 – 4100 Duisburg 11 – W-Germany Tel./Fax 0203/501274

QL-Emulator (Hardware, QJUMP's Drivers AtariDOS and utilities) **£166**
EPROM Cartridge for ST (makes machine autoBOOT) switchable **£33**
Floppy Adaptor to connect QL-Disk-drives to the ST **£9**

QDesign – Graphics program for the Pointer Env. Editable area ranging from 512x256 pixels up to a complete A4 in resolutions up to 300x300dpi (depending on RAM). You can load mono 640x400 ST pictures and use them. **£38**
LaserVersion for Laserjet II, Inkjet, Deskjet £45

FLP/RAM Level 2 - Replacement EPROM for SupeQBoards (V1.17 onwards, with & without mouse) and all TrumpCards. No soldering! New Manual! Real sub-directories! QL-Emulator compatible! About twice as fast, with improved slave-block-handling! **£14**

QD III! Very comfortable text-editor, running under the Pointer Environment. Many features (e.g. Menu, Scrap) **£38** Upgrade from VI or VII - **£10**

QPTR - Toolkit for Machine Code and BASIC Pointer Environment Programmers, including CONFIG. **£30** Update (new manual & disc) **£14**

QMENU Many QD-Users will know it: the Menu Extension. Now with more facilities, even real subdirectories are treated in a new file-select window. Programming instructions and examples for SuperBASIC and machine code. A Scrap Thing is also implemented now!. **£11**

EASYPTR II - Create your own Pointer Environment menus & sprites and use them in your own SuperBASIC or machine-code programs. Many users waited for this package, as creating Menus was not easy, until now! Put all the items you need on screen, that's it! Supports all the PE facilities, even split sub-windows and uses the Menu Extension! Now even complete application-cub-windows may be created! Many examples! **£49**
EASYPTR II - Budget Version **£29**
Upgrade from Version 1 **£10**

QDOS Reference Manual - This book is a must for all m-code programmers. It explains how to use QDOS, all traps and vectors, the Thing System, the HOTKEY System II and much more. It points out which features work on a QL, an Emulator and how to write compatible for future operating systems. DIN A5, 170 pages **£30**

QSUP £26 - QLQ £21 - Thing & EPROM Manager £18.50 - QPROMMER £19
DIAMONDS £11 - Firebirds £10 - BrainSmasher! £12 - Arcanoid II £10
QIMI - Mouse-interface with Real-Time-Clock **£41** without RTC **£37.50**
High Quality Mouse for QIMI or ATARI ST **£27**
HD 3.5" Disc Drive 1.44MB and 720k (just the bare drive, without PSU & Cables, e.g. to replace an old drive to be used with the GoldCard) **£37**

Please add £4.00 for postage and package.
All programs except Arcanoid & Firebirds need memory expansion.
All prices excl. V.A.T. E&OE. Write for more information!

TF Services

MINERVA

The ULTIMATE operating system upgrade

MKII MINERVA now with battery for 256 bytes ram, CRASHPROOF clock & interface (I^2C) for robots etc. Can autoboot from battery ram.

Other features (common to MKI/MKII):
Autoboot on reset/ Multiple Basic/ Faster scheduler-graphics(10% of Lightning)-string handling etc/ WHEN ERROR/ Second screen/TRACE Foreign kybd drivers/ 'Warm' fast reset etc

UPGRADES 1.80+ FREE, 1.79 & earlier - £5
send disk/2 mdvs+ROM: MKI to MKII upgrades

Original MKI..... £40 MKII.....£65

MKI GOLD CARD COMPATIBLE (call for MKII update)

QL SPARES

Faulty QL boards (excl plug in ICs).....	£9		
Keyboard membrane...	£8	Circuit diags...	£2
68008 cpu.....	£8	1377 PAL.....	£3
JM rom set.....	£12	Power supply...	£12
8302 ULA.....	£10	8301 ULA.....	£10
8049.....	£8	MDV ULA	£12

Other components (sockets etc)... pse PHONE

BULLETIN BOARD

QBBS - UK's first QL scrolling board
QL Echomail around Europe, and FIDO netmail
Around 10mb of files including the WHOLE King James' bible!

071-706 2379 V21/23/22/22bis 24hrs

Qualsoft Terminal

Viewdata/VT52/VT100 QL TERMINAL EMULATOR

Multitasking for electronic mail, PRESTEL etc, CET download. Compatible with Tandata Qconnect QL modem/modaptor. Use Qconnect RS232C to drive unbuffered modems. Comprehensive A5 manual.

Software (3.5" or mdv) & manual £30

QL REPAIRS

QLs tested with Thorn EMI test rig and ROM software
(MDV hardware extra if required)

£27 - 6 month guarantee

COMPUTER CLEANER

HELP STOP POWER SUPPLY INDUCED LOCKUPS AND PROTECT COMPUTERS

Much improved mains filters. 40-80 db cut. Contain inductance, spike filters (three in 3 & 4 way) and certified 240vac capacitors.

2-way (5a adaptor).....	£14
3-way (5a adaptor).....	£18
4-way (13a trailing socket).....	£24

All prices include VAT and postage in the UK

FILE TRANSFER

Programs to transfer text AND programs error free (XMODEM) between computers

Available for IBM PC and compatibles, ATARI ST and Sinclair QL. Connects to Psion organiser via Psion comms link

Qualsoft program (per m/c).....	£7.50
Serial lead (name 2 computers)...	£10
Complete package for 2 computers.	£25

MAIL ORDER ONLY

12 Bouverie Place, London W2 1RB (tel: 071-724 9853)
SCROLLING BBS FOR ORDERING/QL HELP - 071-706 2379 (All speeds to V22bis)
Fax: 071-706 2379 Prestel: 017249853

C.G.H. SERVICES

Cwm Gwen Hall, Pencader, Dyfed, Cymru, SA39 9HA
(Tel. 0559 384574) (9am - 5pm)

ALL PRICES INCLUDE MEDIA AND POST AND PACKING FOR THE UK. PLEASE ADD 10% FOR ORDERS IN EUROPE; 20% OUTSIDE OF EUROPE. PLEASE INDICATE 3.5"/ 5.25"/MDV VERSIONS WHEN ORDERING.

COMMERCIAL SOFTWARE

ANELPUM QUAT (NICK WARD) (text adventure) (128k) (flp/mdv) £10.00
ASSAULT AND BATTERY (DAMON CHAPLIN) (shoot 'em up) (128k) (flp/mdv) £12.50
ASTRO (NICK WARD) (astrology prog) (128k) (flp/mdv) £12.50
THE BLAG2 (TONY WOOLCOCK) (text adventure) (256k) (flp/mdv) £10.00
D-DAY MKII (DDC & RICH MELLOR) wargame) (256k) (flp) £15.00
DOUBLE BLOCK (FRANCOIS LANCIAULT) (arcade game) (128k) (flp/mdv) £10.00
DREAMLANDS (JEAN-YVES ROUFFIAC) (text adventure) (256k) (flp) £10.00
FIVE GAMES PACK 1 (WREFORD DAVIES) (mind games) (128k) (flp) £12.50
FROM THE TOWER OF VALAGON (ALAN PEMBERTON) (text adventure) (128k) (flp/mdv) £10.00
GEE-GEE SYSTEM (PHIL JONES AND ANDY CSERBAKOI) (race predictor) (128k) (flp/mdv) £10.00
GREY WOLF (OLIVER NEEF) (u-boat simulator) (256k) (flp) £10.00
HERE WE GO (PHIL JONES AND ANDY CSERBAKOI) (text adventure) (128k) (flp/mdv) £10.00
MACSPORRAN'S LAMENT (DAVE WATSON) (illustrated text adventure) (128K) (flp/mdv) £10.00
OPEN GOLF (OLIVER NEEF) (golf simulator) (384k) (flp) £12.50
ORBITING STARS (JOHN TOPHAM) (astronomical simulator) (128/256k) (flp/mdv) £10.00
PERSONAL FINANCE MANAGER (JASON VICINANZA) (budget system) (128k) (flp/mdv) £10.00
POLYTEXT (NICK WARD) (multi-column quill) (128/256k) (flp/mdv) £17.50
PUDGE (DAMON CHAPLIN) (arcade game) (128k) (flp/mdv) £12.50
QUICK MANDELBROT (KENNETH MURRAY) (fractals) (128k) (flp/mdv) £10.00
QUICK MANDELBROT & JULIA (KENNETH MURRAY) (fractals) (128k) (flp/mdv) £12.50
QUIZMASTER (PHIL JONES AND ANDY CSERBAKOI) (quiz game) (128k) (flp/mdv) £10.00
QUIZMASTER II (AS ABOVE & RICH MELLOR) (quiz game + module option) (128k) (flp/mdv) £12.50
QL EDIT ADVENTURE (ANDY PRITCHARD) (ILLUSTRATED ADVENTURE) (256K) (FLP) £12.50
RETURN TO EDEN (OLIVER NEEF) (role-playing illustrated adventure) (256k) (flp) £17.50
SECTOR X (HORST SPIERLING) (shoot 'em up) (256k) (flp) £12.50 (mdv) £15.00
SPEEDFREAKS (DAMON CHAPLIN) (car race game) (128k) (flp/mdv) £12.50
SQUIDGY ROUND THE WORLD (MICHAEL CROWE) (arcade game) (flp) £12.50 (mdv) £15.00
STARPLOD (ALAN PEMBERTON) (icon-driven adventure) (128k) (flp/mdv) £10.00
UNCLE LOONIE'S LEGACY (DAVE WATSON) (puzzle adventure) (128k) (flp/mdv) £10.00
VOYAGE OF THE BEANO (ALAN PEMBERTON) (illustrated text adventure) (256k) (flp) £12.50
WRECK DIVE (NICK WARD) (arcade adventure) (128k) (flp/mdv) £10.00

CLIP ART 1 - SPORTS (FLP) £6.00
CLIP ART 2 - WHIMSIES (FLP) £6.00
CLIP ART 3 - OFFICE (FLP) £6.00
CLIP ART 4 - VIZ (FLP) £6.00
CLIP ART 5 - GENERAL (FLP) £6.00

SPEEDSCREEN - ROM VERSION £20.00
SPEEDSCREEN - FLP/MDV £10.00

Z 88

EaZy Money	£71.00	Centronics Interface	£31.00
Z-Base	£71.00	128K Eprom Pak	£53.00
Z-Tape	£51.50	128K RAM Pak	£53.00
Z-Term	£51.50	Z88 Computer	£236.00
ThinkZ	£51.50	512K Ram Pack	£202.00
Superchip 4	£154.00	256K Eprom Pack	£58.50
Golf	£25.50	1Mgb Ram Pack	£237.50
Spellmaster	£60.50	Squeeze 88	£48.00
Cambridge Topper	£10.50	Superchip 3	£133.50

PSION ORGANISER

Model XP2	£113.00	Psion Printer	£236.00
Model LZ	£154.00	Spelling Checker	£30.00
Model LZ 64	£195.00	32K Datapak	£36.00
Centronics Interface	£41.00	128K Datapak	£102.50
Comms Link	£61.50	32K DRam	£57.00

TK COMPUTERWARE

the QL stockist

Tel: 0303-81-2801

HARDWARE

768K Trump Card	£230.50
256K Trump Card	£138.50
768K Trump Card and Dual 3.5" Disk Drives	£384.50
256K Trump Card and Dual 3.5" Disk Drives	£292.00
256K Trump Card and Single 3.5" Disc Drive	£215.50
Centronics Interface	£30.00
Disk Card	£102.50
Keyboard-90 Interface	£77.00
QL Scanner	£143.50
QUIMI Interface with mouse	£101.50
Tandata QCOM set	£94.00

SPARES

Keyboard Membrane	£8.50
JS ROM set	£31.00
ZX 8301	£15.50
ULA ZX8302	£10.50
MAB 8049	£7.50
2G 007E	10.50
MC 68008	£26.50
M/drive assembly-(complete)	28.00
Microdrive Assembly (exchange)	£21.50
Voltage regulator (high output)	£5.00
QPower Regulator	£25.50
QL Test kit	£14.50
Repairs	phone

SUNDRIES

VDU cleaning kit	£7.50
Disk Cleaning Kit 3.5"	£15.50
10 Microdrive cartridges	£26.00
Microdrive Cartridges – 4	£13.50
100 Microdrive labels	£4.00
Discs 3.5" 10 DSDD (branded)	£26.00
Discs 3.5" 10DSDD (unbranded)	£11.50
Power Clean Mains Filter	£24.50
Joystick + adaptor	£18.50
Psion 2.35 Set	£23.50
QL Dustcover	£6.00
Printer Ribbons	Phone

SOFTWARE

BUSINESS

Cash Trader	£61.50
Project Planner	£30.00
Sales Invoice	£20.50
Small Traders Pack	£25.50
Stock Accounting System	£41.00

TEXT

Master Spy	£31.00
Page Designer 2	£36.00
Perfection	£80.00
Perfection +	£125.00
Professional Publisher	£90.00
QTYP 2	£31.00
SPY	£15.50
Turbo Quill Plus	£14.00
The Editor	£30.00
The Editor Special Edition	£50.00

FINANCE

Home Finance [Buzz]	£22.50
Investment Monitor	20.50

CAD

Eye Q	£40.00
GraphicQL Plus	£25.50
PCB 1	£102.50
PCB 2	£51.50
TechniQL	£51.50
Technikit	£25.50
The Painter	£51.50
3-D Designer	£39.00
3D Precision	£50.00

PROGRAMMING

Assembly Workbench	£25.50
Basic ally	£20.50
Better Basic	£25.00
Digital C	£30.00
Digital C Special Edition	£50.00
Forth	£31.00
IDIS	£25.00
IDIS Special Edition	£40.00
Macro Assembler - GST	£39.00
Macro Assembler - Talent +	£20.50
Pro Fortran - 77	£101.50
Pro Pascal	£101.50
QMON 2	£20.50
Lightning Special Edition	£60.00
Locksmith	£15.50
Mailfile	£15.50
Media Manager	£25.00
Media Manager Special Edn	£50.00
QFlash Ramdisc + t'kit	£19.50
QFlash Ram Eprom	£25.50
QFlash Ram Utilites	£10.50
QPAC 1	£23.50
QPAC 2	£51.50
Q Top	£46.00
RPM	£15.50
Sidewinder Deluxe	£20.50
Super Toolkit 2 (Eprom)	£30.00
Super Toolkit 2 – Reconfigurable	£30.00
Success	£40.00
Terminal	£31.00
Touch Typist	£12.50
Ultraprint	£20.00
XRef 2.00	£20.50
4 Matter	£15.50

LEISURE

Bridge Player 2	£15.50
Cosmos+	£15.50
Flight Simulator	£19.50
Matchpoint	£15.50
Micro Bridge	£40.00
Patience	£18.50
Prof Astrologer + Astronomer	£80.00
Prof. Astronomer	£40.00
Snooker	£13.50
Strip Poker	£15.50
Super Astrologer	£25.00
Super Backgammon 3	£13.50
Type 22	£18.50

GAMES

Aquanaught 471	£19.50
Arcanoid	£15.50
BJ in 3D land	£12.50
Brain Smasher	£18.50
Crazy Painter	£13.50
Crown of Sorkinen	£18.00
Deathstrike	£15.50
Firebirds	£15.50
Hopper	£13 50
Hoverzone	£15.50
Jungle Eddy	£15.50
Karate	£15.50
King	£13.50
Land of Havoc	£19.50
Lost Pharoah	£15.50
Mortville Manor	£15.50
Stone Raider	£13.50
Super Arcadia	£10.50
Vroom	£13.00

UTILITIES

Archivist	£31.00
Archivist MP	£41.00
ArchRTM+ArchDEV	£20.50
Cartridge Doctor	£18.50
Conqueror	£100.00
Conqueror MS-DOS 5.00	£180.00
Conqueror with DR-DOS 5.00	£180.00
Discover	£31.00
Lightening	£30.00
Super Forth + Reversi	£40.00
Super Sprite Generator	£31.00
Supercharge Special Edition	£25.00
Turbo	£80.00

• Please telephone for details of other products not listed here •

BOOKS

QDOS Companion	£9.00	Z88 Pipedream	£16.00
Assembly Language Prog	£9.00	Using Your Z88	£10.00
Z88 Dabhand Guide	£16.00	Z88 Computing	£10.00
Z88 Ref Manual	£16.00	Z88 Software DLV Guide	£40.00
Z88 Magic	£16.00	Using Psion Organiser	£10.00
		Psion Handling Book	£15.00

ALL PRICES INCLUDE VAT. U.K. POSTAGE £2.50 PER ORDER. WE EXPORT WORLDWIDE

Phone for details of other QL products OR send SAE for our latest catalogue

Credit card phone orders accepted, or send cheques, postal orders, Eurocheques, to:

TK COMPUTERWARE, STONE STREET, NORTH STANFORD, ASHFORD, KENT TN25 6DF

TELEX 966676 PMFAB G

VISA · Access · MasterCard · AMERICAN EXPRESS · EUROCARD

FAX 0303 812892

QL SCENE

Two new lines from star printers

Star Micronics UK has announced two new printer lines. The LC-20 is dubbed 'the CL-10 for the 90s'. A new 9-pin entry-level dot matrix printer in Star's Business Series, the LC-20 is aimed at small businesses, home users, education and small departments in large companies. It has eight near-letter-quality fonts including italics, print speeds of 180 cps in draft and 45 cps in nlq (12 characters per inch), a 4 KB print buffer, paper parking, push tractor feed with short form tear-off, a built in parallel centronics interface and Epson and IBM emulation. An optional serial interface is available.

The LC-20 has a robust case and quiet mode printing, and is priced £199 exclusive of VAT. The LC-200, launched in September 1990, has the same specifications plus colour printing, for £259 ex. VAT.

Star have also launched a range of four 9- and 24-pin printers with colour facilities as standard. Priced between £399 and £599, the Pro-to-Col series replaces the Professional Series, and includes options such as buffer expansion to 236 KB, high speed draft mode, up to 14 fonts, 80 column model and wide carriage model. The Pro-to-Col is designed for 'general high quality office print needs'.

For more information, contact **Star Micronics UK, Star House, Peregrine Business Park, Gomm Road, High Wycombe, Bucks HP13 7DL. Tel. 0494 471111.**

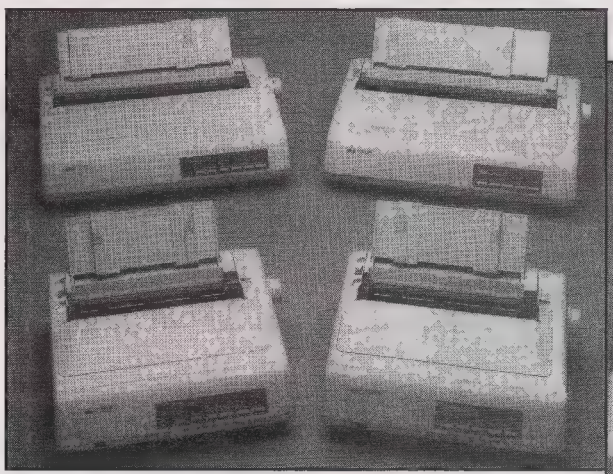
Price changes at CGH

CGH Services have issued a new price list from July 1991. The main points of interest are that they are no longer supporting 'base' prices, where purchasers send their own media. This applies to the public domain library as well as commercial sales. The public domain library is no longer supporting mdvs, and will charge £2 each (including UK postage and packing) for disks supplied. 'We'll pack them as tight as we can,' stresses Richard Alexander.

All commercial prices will now have UK postage included. There is no VAT, as CGH are below the VAT threshold. Non-UK customers add 10% (Europe) or 20% (elsewhere) to cover postage.

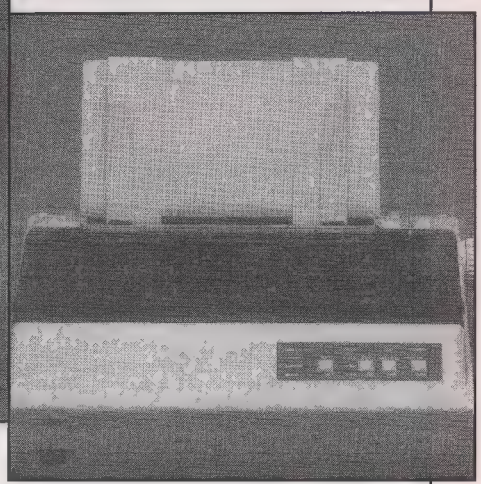
CGH are also revising their authors' royalties structure to remove anomalies.

For a price list contact **CGH Services, Cwm Gwen Hall, Pencader, Dyfed, Cymru SA39 9HA. Tel. 0559 384574.** Send an SAE or international reply coupons for their full public domain catalogue.



Above: THE STAR PRO-TO-COL SERIES

Below: THE LC-20



Cowo release new QTop version

Cowo Electronic have produced the first major revision of their user front-end package *QTop*, the more powerful VI.10. The full QTop system is based on six stand-alone executable multitasking programs. QTop-Desk is the system manager, offering main menus for Desk, Devices, Files, Jobs, Keyboard, Programs, Myprogs, Tools and Options, with over 110 submenus. Cowo believe that the Files and Jobs handling routines are 'the most powerful and complex in the whole QL world.' Real sub directories on all QJump Level 2 device drivers (Miracle Hard Disk, *Gold Card*, Atari ST/QL emulator) and on Thor hard disks are supported.

QTop-Clocks is a multi-functional clock program offering four different digital clocks, an analogue clock and an alarm clock. QTop-Demos offers ten different high-speed graphics demonstrations. QTop-Animator is a flexible graphics animation program. QTop-index tests the hardware of your computer system. QTop-Snap is a screen dump and save program. The latest QJump Pointer Environment is included — it is not necessary to run the Pointer Environment, it is fully supported if installed.

QTop is fully user-configurable, and includes a screen refresh and freely moveable and resizeable windows.

The software with a 41-page manual is available in English or German for £35 (on 3.5 or 5.25 in disk) or £39 (on three microcassettes), prices inclusive of post and packing.

For a six-page information brochure and order form, contact **Cowo Electronic, Urs Konig, Munsterstrasse 4, CH-6210 Sursee, Switzerland.**

OPEN CHANNEL

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide somebody with the answer, or just sound off about something which bothers you, write to: Open Channel, Sinclair QL World, 116/120 Goswell Road, London EC1V 7QD.

Thank you

As a fairly average QL user, most of my purchases over the years have been promoted by articles and reviews in your pages. As a result of Mark Knight's May 1991 review, I recently bought *Perfection* and have found it to be the definitive answer to all *Quill's* shortcomings, *Pro Publisher*, with perseverance, can be splendid in its own special field, but Perfection (unlike other attempts to replace Quill) is child's play to the relatively inexperienced wp user, yet it is still capable of great sophistication. It is, quite simply, marvellous, especially when coupled with *Lightning*. Thank you for Perfection, and thank you, too, for telling me about Pro Publisher, the elegantly simple multi-tasking Taskforce, and my splendid (and cheap) LC24-10 printer.

Brendan Connors
Hove
Sussex

Editor's comment: That's why we run full-length user-driven printer reports every so often.

Listing

Can anyone help me with a program for listing my SuperBasic programs? I have a C-Itok daisywheel printer and have tried the following program, which does in fact provide a carriage return after each line.

Baud 9600: Open #3, serl: list #3: close #3

Result:

10 Rem prog list
20 next line
30 next line
40 next line
50 next line

etc.

This listing actually runs off the page and is lost, leaving me very frustrated. The printer works normally with *Quill* and *Abacus*. Any suggestions from readers would be appreciated.

Malcolm Roberts
Pembroke
Dyfed

Thank you 2

I should like to express my gratitude through your column for the excellent service and expert help given to me by Mr. Freddie Vachha of Digital Precision Ltd. I had been using a single 3.5 in disk drive with a 768K Trump Card on my QL and had ordered from them copies of *QFlick* and *Perfection*. On reciept, neither would load, and the error message 'not found' was given.

On the telephone, Mr. Vaccha advised (a) cleaning and (b) realigning the reading heads. Cleaning proved ineffective, but I had on order from Miracle Systems a dual 3.5 in drive unit, and when this arrived and was connected, *Perfection* loaded and operated perfectly.

In addition to his 'remote diagnosis' of my problem, Mr. Vaccha suggested a firm who would re-align my single drive unit, and I feel his expertise is largely instrumental in keeping alive an excellent computer.

I also greatly appreciate your series *The New User Guide* for people who only have elementary programming skills, and would like to become more proficient.

I receive *QL World* by annual subscription, and have completed the QLAW questionnaire.

R A Mills
Heighington
Lincoln

Positive

The very positive review of *Perfection*, Digital Precision's new word processor, in these pages encouraged me to order what appeared to be a very worthwhile package.

I have now received it and find it excellent, easy to use and the answer to my wordprocessing problems.

It is a pleasure to find software whose performance exceeds even the suppliers' advertising blurb.

J R Haldane
Gorebridge
Midlothian

Interest

This concerns the reader's problem in the August issue under the heading 'Capital'. I have two programs which I contributed myself, which I find very useful, and I hope Mr. Patterson will find them useful too. One is for *Abacus*, while the other is a simple SuperBasic program.

The Abacus program gives accrued interest for a year of 12 equal months, which results in

Editor's notebook

There are ways you can tell that the recession is biting. House-owning colleagues speculate on the possibility of selling up and moving to somewhere rented. Holidaying colleagues are under canvas in Wales instead of under blue skies in Greece. One well-off acquaintance returned from Greece moaning that he wished he'd stayed there. Unusual? You bet. Normally you can't lever him away from his whizz-o-phone.

He works for a big – I mean *very* big – computer company. Their business isn't what it was. Oddly, although we hear the occasional lament that things are slow, QL business seems to be moving along steadily.

Normally the long, hot days of August are the cue for silence from the outside world. Not so this year – QLers are sitting at home writing! So we have the first of a new series of machine code programming, a handful of reviews and a monster DBQL, as well as the regulars.

The Formulae used

```
    ! A !B!  C  !          D          !E!  F  !G!          H          !I!    J    !K!    L    !
    ----------------------------------------------------------------------------------------------
  1!1991* Month!Capital                ! rate% ! accrued              ! In/out(-) *    Growing
  2!    * ==========================================================================*    Accrued
  3!    * B/f  !                  £0.00 !       !                      !           * ==========
  4!    * Jan  !D3+H4+J4                ! 0.00  ! (D3+J4)*F4/1200      !           * H4
  5!    * Feb  !D3+sum(H4:H5)+sum(J4:J5)  ! F4    ! (D3+J4+J5)*F5/1200    !           * L4+H5
  6!    * Mar  !D3+sum(H4:H6)+sum(J4:J6)  ! F5    ! (D3+sum(J4:J6))*F6/1200  !           * L5+H6
  7!    * Apr  !D3+sum(H4:H7)+sum(J4:J7)  ! F6    ! (D3+sum(J4:J7))*F7/1200  !           * L6+H7
  8!    * May  !D3+sum(H4:H8)+sum(J4:J8)  ! F7    ! (D3+sum(J4:J8))*F8/1200  !           * L7+H8
  9!    * Jun  !D3+sum(H4:H9)+sum(J4:J9)  ! F8    ! (D3+sum(J4:J9))*F9/1200  !           * L8+H9
 10!    * Jul  !D3+sum(H4:H10)+sum(J4:J10)! F9    ! (D3+sum(J4:J10))*F10/1200!           * L9+H10
 11!    * Aug  !D3+sum(H4:H11)+sum(J4:J11)! F10   ! (D3+sum(J4:J11))*F11/1200!           * L10+H11
 12!    * Sep  !D3+sum(H4:H12)+sum(J4:J12)! F11   ! (D3+sum(J4:J12))*F12/1200!           * L11+H12
 13!    * Oct  !D3+sum(H4:H13)+sum(J4:J13)! F12   ! (D3+sum(J4:J13))*F13/1200!           * L12+H13
 14!    * Nov  !D3+sum(H4:H14)+sum(J4:J14)! F13   ! (D3+sum(J4:J14))*F14/1200!           * L13+H14
 15!    * Dec  !D3+sum(H4:H15)+sum(J4:J15)! F14   ! (D3+sum(J4:J15))*F15/1200!           * L14+H15
 16!    ##########################################################################################
```

A building society account on Abacus

Some Figures

```
    ! A !B!  C  !    D    !E!  F  !G!   H   !I!    J    !K!    L    !
    ---------------------------------------------------------------------
  1!1991* Month!Capital    ! rate% ! accrued ! In/out(-) *    Growing
  2!    * ==================================================*    Accrued
  3!    * B/f  !£21008.32 !       !         !           * ==========
  4!    * Jan  !£21175.69 !  9.56  !  167.37 !           *     167.37
  5!    * Feb  !£22351.02 !  9.56  !  175.33 !      1000 *     342.70
  6!    * Mar  !£22509.11 !  8.62  !  158.09 !           *     500.79
  7!    * Apr  !£22667.21 !  8.62  !  158.09 !           *     658.89
  8!    * May  !£22246.62 !  7.80  !  139.41 !      -560 *     798.30
  9!    * Jun  !£22386.03 !  7.80  !  139.41 !           *     937.71
 10!    * Jul  !£23531.95 !  7.80  !  145.91 !      1000 *    1083.63
 11!    * Aug  !£23677.86 !  7.80  !  145.91 !           *    1229.54
 12!    * Sep  !£23823.78 !  7.80  !  145.91 !           *    1375.46
 13!    * Oct  !£23969.69 !  7.80  !  145.91 !           *    1521.37
 14!    * Nov  !£24115.60 !  7.80  !  145.91 !           *    1667.28
 15!    * Dec  !£24261.52 !  7.80  !  145.91 !           *    1813.20
 16!    ##################################################################
```

a 360-day year. This will be close enough for most purposes. The interest rate will have to be amended each time it changes and each time that happens the program automatically amends the rate for the remaining months of the year, because each cell in the column uses the reference of the one above it. Thus, one has a forecast of one's position for the year-end.

The accrued interest figures are meant to be taken as at the close of business on the last day of the month in question; the last day of December is therefore the last day of the year.

Deposits and withdrawals are entered in the 'in/out' column, but in order not to make the program unnecessarily complicated, no provision has been made in the column to stipulate on which day of the month the transaction was made.

Each row can be typed one at a time, or one can go as far as March and then COPY the row for March to get April, editing month-name and cell references where necessary, ensuring that D3 and J4 are referred to in every line thereafter. Likewise, one can COPY March and April to get June and July - and March, April, May, June and July to get August, September, October, November and December.

Once I was sure that I had the programs I wanted, I made a COPY of the whole thing twice on rows below the ones in use, and made the figures in them zero. This is a useful procedure, because one cannot LOAD a second file on to the spreadsheet, and keep the one on it which is already in use, as far as I know. With the procedure one can always COPY a blank for next year onto the rows below.

The SuperBasic program will be self-explanatory once it is run.

Peter Tomlin
Hatherley Grove
London

A compound interest table in SuperBasic

```
100 MODE 4: INK 4: PAPER 0:
CLS
110 INPUT "Sum", x
120 INPUT "Years", y
130 INPUT "Rate", t
140 LET r=t/100
150 CLS
160 FOR i=1 TO y
170 LET s=x*((1+r) ^i)
180 PRINT "  Year"!i,, "£"; s
190 END FOR i
200 PRINT
```

Strings

I have found a Bug in Superbasic I have not seen detailed before. It concerns the comparison of strings. Some characters are considered equal by the interpreter, which are obviously not I stumbled on this while writing a guarded input routine; the character O is effectively the same as the cursor up key. The only way to distinguish is to compare CODEs. I have since found a total of 64 cases (see enclosed printout). There is an interesting effect: for example, A is equal to CHR# 225, and a equal also to CHR# 225 but A is not equal to a (using == completes this).

A few months ago Mike Lloyd gave a function for signum: REturn 1-(x<=0)-(x<0). This does have an elegant simplicity, however, the alternative, RET (x>0)-(x<0) works just as well.

Caleb Leeke
Huntingdon
Cambs

```
. CHR$(46)  = CHR$(206)
0 CHR$(48)  = CHR$(208)
1 CHR$(49)  = CHR$(209)
2 CHR$(50)  = CHR$(210)
3 CHR$(51)  = CHR$(211)
4 CHR$(52)  = CHR$(212)
5 CHR$(53)  = CHR$(213)
6 CHR$(54)  = CHR$(214)
7 CHR$(55)  = CHR$(215)
8 CHR$(56)  = CHR$(216)
9 CHR$(57)  = CHR$(217)
A CHR$(65)  = CHR$(225)
B CHR$(66)  = CHR$(226)
C CHR$(67)  = CHR$(227)
D CHR$(68)  = CHR$(228)
E CHR$(69)  = CHR$(229)
F CHR$(70)  = CHR$(230)
G CHR$(71)  = CHR$(231)
H CHR$(72)  = CHR$(232)
I CHR$(73)  = CHR$(233)
J CHR$(74)  = CHR$(234)
K CHR$(75)  = CHR$(235)
L CHR$(76)  = CHR$(236)
M CHR$(77)  = CHR$(237)
N CHR$(78)  = CHR$(238)
O CHR$(79)  = CHR$(239)
P CHR$(80)  = CHR$(240)
Q CHR$(81)  = CHR$(241)
R CHR$(82)  = CHR$(242)
S CHR$(83)  = CHR$(243)
T CHR$(84)  = CHR$(244)
U CHR$(85)  = CHR$(245)
V CHR$(86)  = CHR$(246)
W CHR$(87)  = CHR$(247)
X CHR$(88)  = CHR$(248)
Y CHR$(89)  = CHR$(249)
Z CHR$(90)  = CHR$(250)
_ CHR$(95)  = CHR$(255)
a CHR$(97)  = CHR$(225)
b CHR$(98)  = CHR$(226)
c CHR$(99)  = CHR$(227)
d CHR$(100) = CHR$(228)
e CHR$(101) = CHR$(229)
f CHR$(102) = CHR$(230)
g CHR$(103) = CHR$(231)
h CHR$(104) = CHR$(232)
i CHR$(105) = CHR$(233)
j CHR$(106) = CHR$(234)
k CHR$(107) = CHR$(235)
l CHR$(108) = CHR$(236)
m CHR$(109) = CHR$(237)
n CHR$(110) = CHR$(238)
o CHR$(111) = CHR$(239)
p CHR$(112) = CHR$(240)
q CHR$(113) = CHR$(241)
r CHR$(114) = CHR$(242)
s CHR$(115) = CHR$(243)
t CHR$(116) = CHR$(244)
u CHR$(117) = CHR$(245)
v CHR$(118) = CHR$(246)
w CHR$(119) = CHR$(247)
x CHR$(120) = CHR$(248)
y CHR$(121) = CHR$(249)
z CHR$(122) = CHR$(250)
```

SOFTWARE FILE

THE GOPHER

INFORMATION
Program: *The Gopher* V1.02
Price: £12 (add £1 post age for EEC countries,£2.50 elsewhere).
Supplier: Dilwyn Jones Computing, 41 Bro Emrys, Tal-y-Bont, Bangor, Gwynedd LL57 3YT. Tel: (0248) - 354023.

Bryan Davies investigates a species that tracks down elusive disk files and reports them back to base.

The hard disk user is in a paradoxical situation. It is perhaps the most expensive add-on unit for the QL, but there is little software available for it, to spend more money on. At the time of writing, Dilwyn Jones Computing offer the only two items that users are likely to get to hear about. *WinBack* was dealt with in a previous review; it is a fairly basic hard disk backup program, which works well. *The Gopher* was written by the same person, so one would expect it to be of a similar nature and, indeed, the similarity is clear.

The Gopher is available on microdrive, and SuperBasic and compiled with Digital Precision's *Turbo*. It requires the usual Turbo RUNTIME_EXTS file to be loaded, plus an additional one called GOPHER_BIN, for SB commands specific to The Gopher. If a suitable version of the XTRAS or TURBO_TK_CODE file has already been loaded, the runtime file is not needed; the boot routine asks you whether or not it should go ahead and load it. About 32 KB of memory is required to run the program. In case the origin of the name is not yet clear, the program's function is to 'go for' files, that is, ferret out files which the user cannot (easily) locate. While such a utility obviously appeals most to the hard disk user, because of the much greater number of files this drive holds than either floppy or microdrive, there are some users who manage to get a hundred or so files on their floppies and may have difficulty locating required ones.

Booklet

There is an 11-page instruction booklet, which covers program operations adequately. An UPDATES_DOC file is provided on disk (or cartridge) to bring the purchaser up to date on happenings since the manual was printed. The style is informal, without being chatty, and lets the purchaser in on some of the program development details. The manual includes considerable explanation on using the clone program supplied, but it is easy simply to COPY OR WCOPY the files to a working disk or cartridge. The manual doesn't deal with the purpose of two files on the supplied disk, and you have to read one of those files to find out what the other is for. It is a program for altering the data space taken by the executable files; users of the Turbo compiler will find this quite understandable, but average *Quill* users may find it a complete mystery. There is no mention of why users of The Gopher should need to change data space, so the best thing the user can do is forget it.

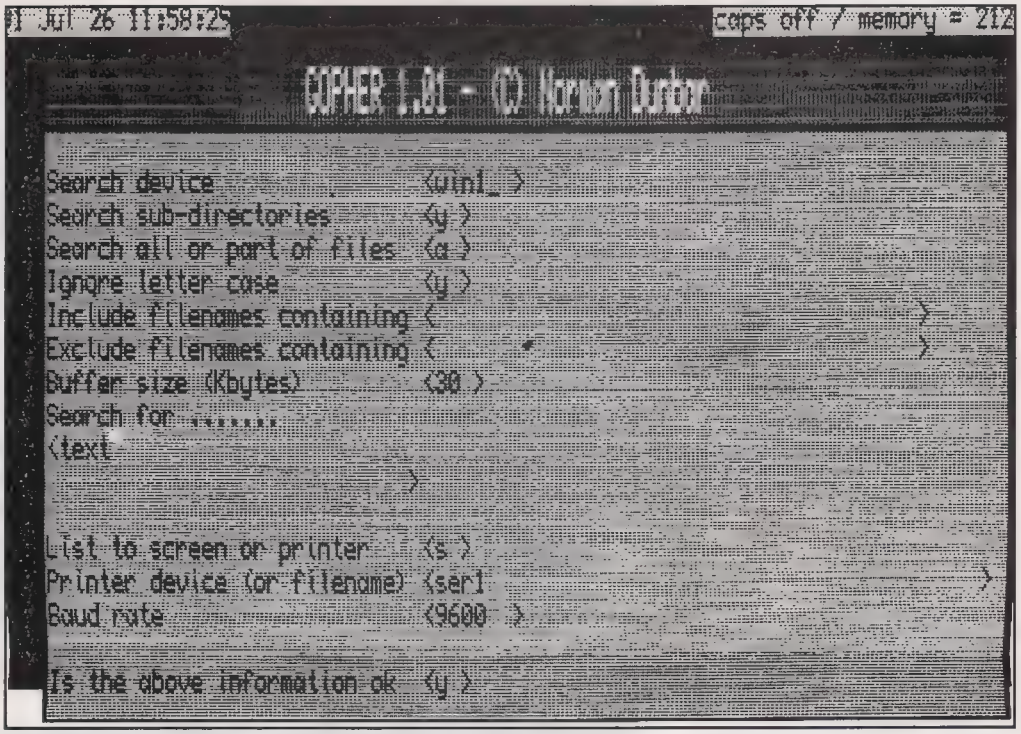
If finding a file were only a matter of tracing a particular file name, it might be indulgent to employ a program to do the job for you. A reasonably organised user will give files names which are not hard to relate to file contents. The flysheet on The Gopher supplied by DJC gives the example of tracing letters written to *QL World* (a fair task, in my case). The restriction of eight characters for a file name means that you can't hope to provide much, if anything, of a clue to the file contents in its name; at best, you can have something like QLWLET1.DOC etc. You'll know whether or not a file is indeed a letter to QL World, but won't have a clue as to a letter's contents. The program must, therefore, be capable of looking *into* files, and checking there for key text strings, from which the file contents can be gauged. It is not presently usual for QL word-processing programs to provide a 'comments' facility, separate from the file text itself, and The Gopher has to go through the text until it finds, or fails to find, any specified character string. With long files, this could be a lengthy process.

To speed-up the search process, it is desirable to exclude any groups of files which are unlikely to contain what is being looked for. Maybe you always write your letters to QL World with *text* [87], in which case you can specify that only files with the _T87 extension in their names are examined. In case the impression given so far is that the program is a tool for word-processing users only, rest assured that it can be used with any files, but string searches are obviously limited to what characters can be entered into the string. The manual doesn't list the acceptable characters, but 'specials' such as 0 can be used. The search string can be up to 100 characters in length. The search used when producing the screen dumps given with this article took about six minutes to complete. There were 192 files on the hard disk, occupying 2.7 MB, and 35 instances of the search string were reported as found. A more specific, case-dependent search, for the string INDEXICAL, took well under one minute, when the file suffix (_WP) of the only file found had been specified, whereas it took seven minutes with no restrictions made.

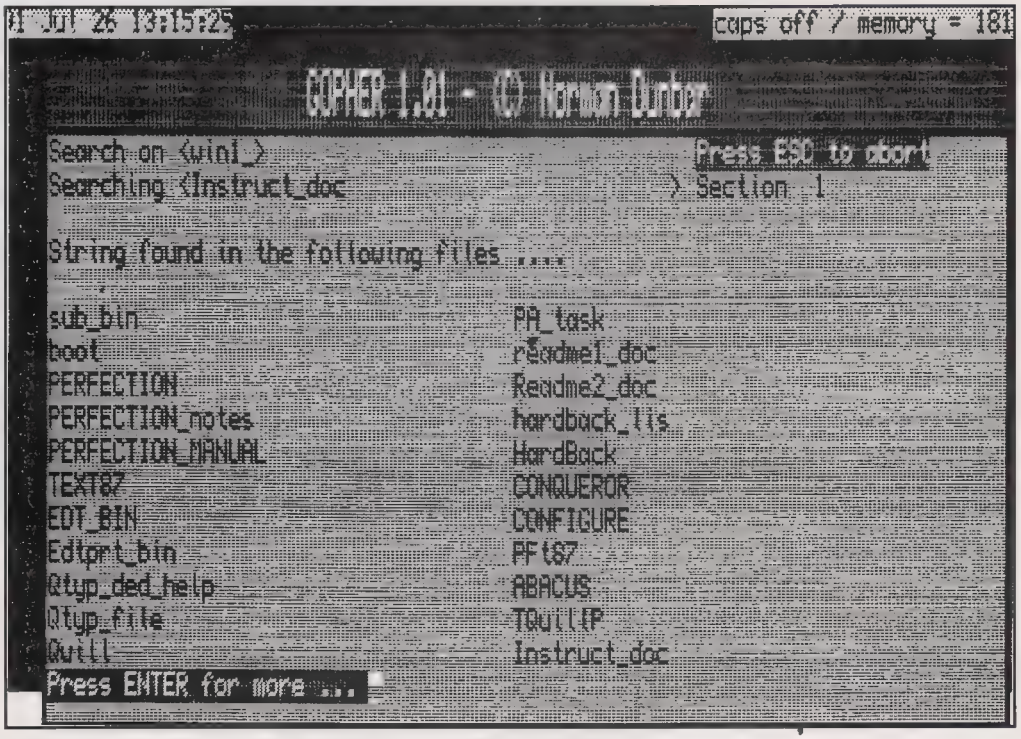
Figure one is of the display at the stage of specifying initial conditions for a search. The range of conditions is greater than one might expect. For in-

stance, you can speed-up a search, if you are interested in finding the search string only in the first few pages of documents, by opting to search only the first part of each file, and specifying a buffer size sufficient just for the pages you want searched. The default buffer size for a part-search is 2 KB. The minimum is 1 KB, nominally equivalent to about 150 words of text (but bear in mind that text files tend to have formatting information at the front and even a file with no text in it may occupy 2Kb).

The buffer action is different if whole files are to be searched; the search is then fastest if the buffer is set to the maximum 30Kb. The program loads files in chunks of the buffer size specified. When one chunk has been examined, the next is loaded. The decision on case-sensitive or not affects search speed, too. Entering <N> here – that is, case is to be taken into account – is stated to reduced the search time by as much as a half.



The display specifying initial conditions for the search.



The search has found the string in a number of files.

Legal

The 'legal' entries for Search Device are any valid QL device with a five-character name; the underline must be the last character, and the drive number must be specified (in the range) 1-8). You cannot enter a sub directory name, therefore. Those users who have successfully constructed an extensive sub directory structure will be glad the option is provided to specify whether or not (all) sub directories are to be searched. It is pointed out in the instructions that you can check all sub directories of a Miracle Systems hard disk by using CTRL-C to switch to SB, then typing <WIN2> and pressing ENTER, then returning to the program. The point here is that the hard disk software may have been set for a particular sub directory, before The Gopher was started, and the program will then not be able to look at the whole disk. At present, the sub directories option is available only if you have entered <WINn> as the Search Device. Presumably, provision will be made before long for the high-density floppy disk drives with the Miracle Systems *Gold Card*, since they have the same sub directory structure available as for the hard disk.

Apart from entering the search string, all you need do much of the time is press ENTER for all entries. When the search string is located, the name of the file it was found in is written to the screen, or to both screen and printer, as specified. If the printer option is selected, you can opt to send the files names shown on-screen to either printer or disk drive file. If there are enough instances of the string being found, you are requested to press ENTER to clear the screen and make way for any further instances. The maximum number of files displayable per screen is 22. **Figure two** shows the display when a search has found more than the 22 instances of the search string noted in the first illustration – <text>. In this case, whole files were searched, with the buffer set to 30 KB. **Figure three** shows how setting the program to do part-searches, with a buffer of 2 KB, affected the number of instances found.

Unlike many small programs, The Gopher does have a refresh key (the standard F4) to clean the screen if another program has corrupted it. Being a multi-tasking program, it is quite likely The Gopher will have company in the QL. In fact, the review was done with

my normal system fully loaded, and this didn't seem to upset The Gopher at all. As the *Perfection* WP program was being reviewed at the same time, it was loaded, as was its accessory printer-driver creation program, *Flashback 2*, *Files 2*, *Q-Switch*, and a *lot* of extensions. Apart from Files 2 disappearing on one occasion (possibly not connected with The Gopher), there was no apparent interaction between programs. Error trapping is incorporated in the program, so you are unlikely to be caught out by things like failing to put a disk in a search drive.

Killed

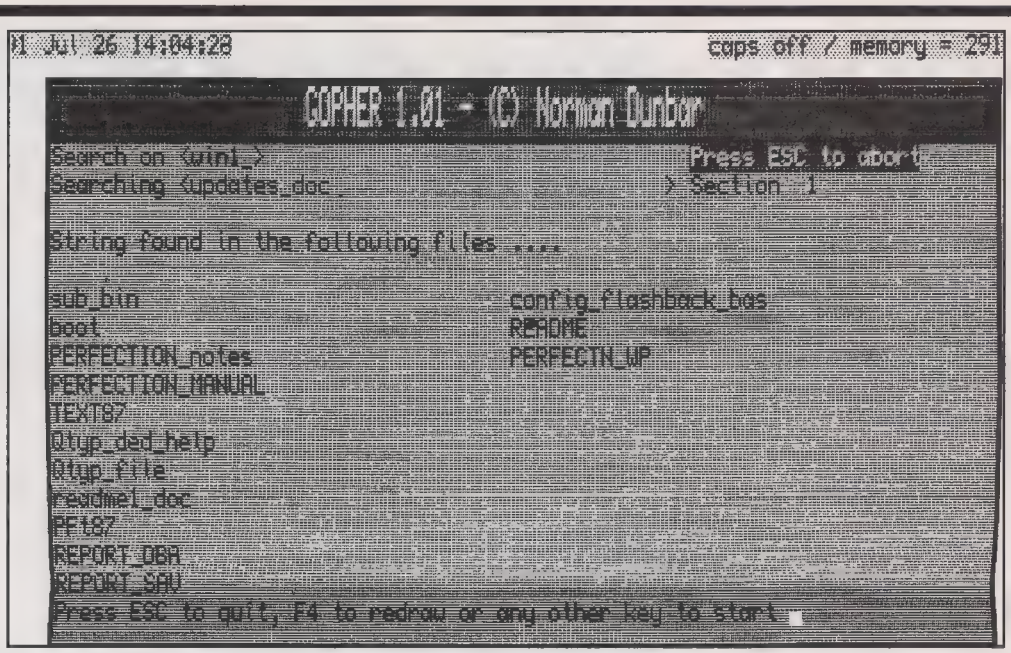
The Gopher clearly did not like one file, namely QTYP _DICTIONARY, which is to be found as part of QJump's *Qtyp* spelling checker and later versions of text[87]. The program stopped when it came across this file on hard disk, and gave the message 'file error'. The option was offered to press ENTER in the hope the search could be continued, but the attempt to resume was unsuccessful and the program had to be killed and restarted. This happened with subsequent attempts too, making the program unusable until the QType dictionary had been deleted from the hard disk. Rather an extreme reaction. However, an earlier version of the file on a floppy gave no trouble and copying that to the hard disk removed the problem. Maybe the file was corrupted in some way; it had been edited, but with the QType dictionary editor.

All files are OPENed to be checked. Users of Archive who have lost files through not having them CLOSED successfully might be made nervous by the thought of so many files being opened, but there may be no alternative to doing this. On the matter of what characters can be entered as part of a string, some users might wish for non Ascii characters such as the alphabetic capitals with a bar over the top, as used by *The Editor*.

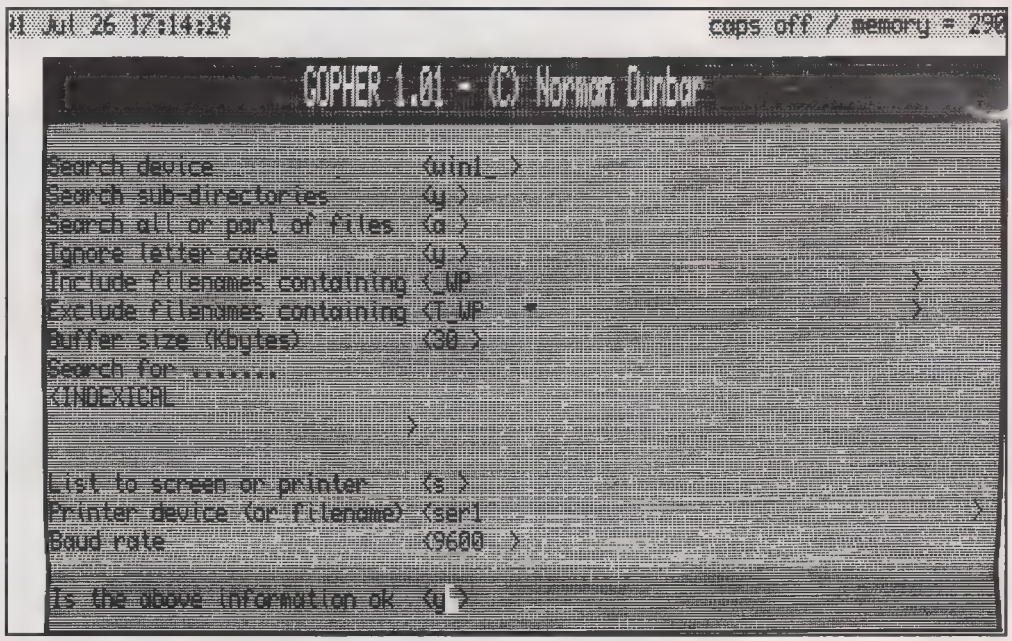
Neither DJC nor the writer of the program go over the top in making claims for it, nor do they ask a lot of money for it. There are some features one would like to see incorporated into it, but it would be only reasonable to expect the price to go up if they were added. One obvious wish is for a display of the location of a string, when it is found. No doubt adding such a feature would be time-consuming, both in the programming required and in the effect it would have on the speed of the program, but it might save having to load into several files which were found to have the desired string in them, in order to discover which instance is the required one.

A smaller wish is that the search string should be displayed during the search. There are some of us who tend to forget what we have done, very quickly; it is also desirable to be able to check whether or not one put spaces (for instance) in a string. The instructions could have been clearer on the matter of how Include and Exclude entries can be typed in; there was space for 36 characters, which is the Qdos file name limit but is more than would be used in a normal file name (12 maximum). Presumably, only single, complete strings can be used. It would be handy to be able to enter several strings – eg _DOC _EXT _T87 _BIN – and have them all, individually, excluded/included. **Figure four** shows how these entries are actually used. All files having <T_WP> as part of their names would not be checked, but *any remaining* files having <_WP> would be checked. This excluded *all* files which did not have <_WP> in their names, and also excluded any files which had <T_WP>.

The Gopher is a fairly basic program, without pretensions to greatness, but it is straightforward to use, and works well. There is the possibility of additional features being added in upgrades; presumably this will depend upon how well the program sells in its current version. At the price, it is good value.



The results of a part-search with a 2 KB buffer.



The display showing the include/exclude options.

QL SCENE

Solent sub-group launches new Quanta workshop

The Solent sub-group of Quanta is holding its first Workshop on Sunday 22 September from 10am to 6pm at the Horizon Centre, Sundridge Close, Portsmouth. The modern venue has 'excellent facilities', including facilities for the disabled, and plenty of car parking. It is close to the A3 and the M27 and is also handy for cross-Channel ferries from Europe. Food and drink will be available all day, and the licensed bar will be open at lunch time.

There will be a number of lectures or presentations on a wide range of QL-related topics by the well-known and not so well-known. Traders will be present and there will be new products to see and assess. There will be facilities to get together and discuss recent new developments in the QL market.

The organisers also emphasise that these events are called 'workshops' for a good reason – they encourage attendees to bring their QLs and relations along, 'set them up and have a good time'. There will be a Clinic Desk for problem-solving, and competitions and games for youngsters, who are also welcome.

For information about the event, transport and accommodation needs, etc., please contact **Graham Goodwin 0489 895451, Graham Evans 042121 3350 or (regional transport, accommodation and other local events) Portsmouth Tourist Information office 0705 838382.**

Other workshops

Other Quanta workshops planned for the near future are **Birmingham** (28 September) at the Sandwell District General Hospital Postgraduate Centre (call Bill Fuggle 021 4726671 or Dennis Briggs 0952 255895 for information), **Nottingham** 12-13 October (no further information given) and **Bristol** 17 November (no further information given).

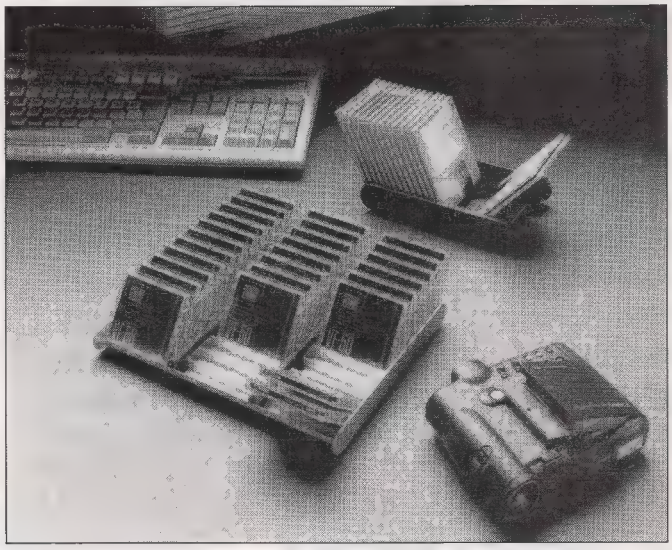
Fairing further

The All Formats Computer Fair has extended its range again. After two successful fairs in the Midlands, the All Formats Fair is now happening at five venues around the UK. At least four visits to each venue a year are planned. The next few fairs are:

Midlands: National Motorcycle Museum, Solihull, M42 (opposite the National Exhibition Centre and Birmingham International BR station), Saturday 14 September; **Scotland**: City Hall, Candleriggs, Glasgow, Sunday 22 September; **Western**: The Brunel Centre, Bristol Old Station (near Temple Meads Station), Sunday 6 October.

The admission charge is now £4 but the stand price has dropped to £60. All fairs start at 10am and finish at 4pm. All venues have ample car parking. For advance tickets and stands, contact **John Riding on 0225 868100.**

Little boxes



Lift UK Ltd, established makers of storage systems for music and video recording, are launching a range of storage units for disks.

The Databoy range includes desk units for 2 in, 3.5 in and 5.25 in disks. The ridged construction means that disks can be flipped forward for browsing, without the row collapsing.

23-disk units will cost £3.99 and 43-disk units will cost £5.99. Models with protective lids are expected to follow. Lift are also advertising a logo-stamping service for promotional purposes.

Information from **Lift (UK) Ltd, Triangle Business park, Wendover Road, Stoke Mandeville, Bucks HP22 5BL. Tel. 0296 615151.**

MIRACLE SYSTEMS

MASTERCARD VISA

QL GOLD CARD

£375 inclusive (£330 export)

• 2 year warranty • 14 day money back guarantee •

- The **GOLD CARD** is the ultimate expansion for the QL. It speeds up program execution by a factor of about four, increases the memory to 1920K bytes and enables the QL to interface to 3.2M Byte disk drives (as well as 720K and 1.44M).
- There is a battery backed clock so that the QL powers up with the correct time and date.
- We offer part-exchange for old expansions:

Part Ex. Item	GOLD CARD price	
TRUMP CARD 2	£225.00	(£200.00 export)
TRUMP CARD 768K	£275.00	(£241.00 export)
TRUMP CARD 256K	£300.00	(£263.00 export)
EXPANDERAM	£325.00	(£284.00 export)
DISK CARD	£325.00	(£284.00 export)
RAM + DISK I/FACE	£325.00	(£284.00 export)
Other Expansion	£350.00	(£305.00 export)

- All the functions on the TRUMP CARD, i.e. TKII, printer buffer, screen dump, RAM disk and memory cut, are carried over to the **GOLD CARD.**
- The floppy and RAM disk drivers now support sub-directories.

JOIN QUANTA !

QUANTA is the independent QL user's club. If you have not already joined then consider the following:

- QUANTA stands for "QL Users and Tinkerers Association" and provides a forum for users to get together.
- Members requiring guidance on software and hardware can seek out expertise from fellow members.
- Local sub-group meetings are regularly held throughout much of Britain.
- There are also workshops where we and others demonstrate our wares. Members have access to a large software library and can often find spare parts and second hand bargains at QUANTA meetings and can see new products.

The £14 (£17 overseas) subscription includes an informative monthly magazine. Most of all, though, it is good fun so contact the Membership Secretary, Bill Newell, 213 Manor Road, Benfleet, Essex, SS7 4JD, or telephone (0268) 754407 now!

Tel: (0904) 423986

To place an order by phone please have your credit card ready. For overseas customers we charge the prices shown in brackets.

To order by post, please fill in the form opposite or write to us quoting your credit card number and expiry date, or enclosing a cheque payable to MIRACLE SYSTEMS Ltd.

To: MIRACLE SYSTEMS Ltd, 25 Broughton Way, Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986

Please send me _____

I enclose a cheque to the value of: £ _____

Or debit credit card ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

expires ☐☐/☐☐

Name _____ Signature _____

Address _____

_____

_____

MIRACLE SYSTEMS

MASTERCARD VISA

QL 3.2 Megabyte Dual Disk Drive

New standard for the QL

£175 inclusive (£155 export)

• 2 year warranty • 14 day money back guarantee •

The **3.2M byte DUAL DISK DRIVE** together with the GOLD CARD brings the QL system right up to date. The drive contains 2 mechanisms each able to store 3.2Mb on an ED diskette. ED actually stands for 'Extra High Density' and refers to the rate at which data bits are read from and written to the disk. ED disks transfer data at 1 Megabit per second whereas HD (High Density) transfers at 500K bit/s and the old QL standard DD (Double Density) at 250K bit/s. Please note that these are instantaneous rates and that to work out file transfer times other factors must be considered.

The drive is fully backward compatible in that it can read and write both DD and HD diskettes with any type in either drive. The DD, HD and ED diskettes are all the same physical size known as the 3.5" form factor. The HD and ED diskettes have an extra hole in their casing, in different places, so that the drive can detect which type of diskette has been inserted. This means that there is no need for the user to tell the QL of the diskette type in use. For example, if you format a DD it will format to 720K, whereas an ED would format to 3.2M.

The drive is fully cased with cable and integral 220/240V power supply and comes with 10 ED diskettes.

We are exhibiting at:

14 Sept:	All Formats Fair, National Motorcycle Museum, Birmingham (Junction 6 off M42).
22 Sept:	QUANTA WORKSHOP, The Horizon Centre, Sundridge Close, Portsmouth.
22 Sept:	All Formats Fair, The City Hall, Glasgow.
28 Sept:	QUANTA WORKSHOP, Postgraduate Centre, Sandwell District Hospital, Birmingham.
6 Oct:	All Formats Fair, The Brunel Centre, Temple Meads, Bristol.
12 Oct:	QUANTA WORKSHOP, Nottingham.

OTHER PRODUCTS:

(more details about our products may be found in the QUANTA Magazine)

ITEM	PRICE	
QL CENTRONICS	£29.00	(£28.00 export)
QL 40M HARD DISK	£449.00	(£405.00 export)
* TRUMP CARD 768K	£100.00	(£89.00 export)
* TRUMP CARD 2	£150.00	(£130.00 export)
* EXPANDERAM 512K	£50.00	(£46.00 export)
5.25" DISK DRIVE	£95.00	(£88.00 export)
DISK ADAPTER	£15.00	(£15.00 export)

* reconditioned product

MIRACLE SYSTEMS Ltd, 25 Broughton Way, Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986

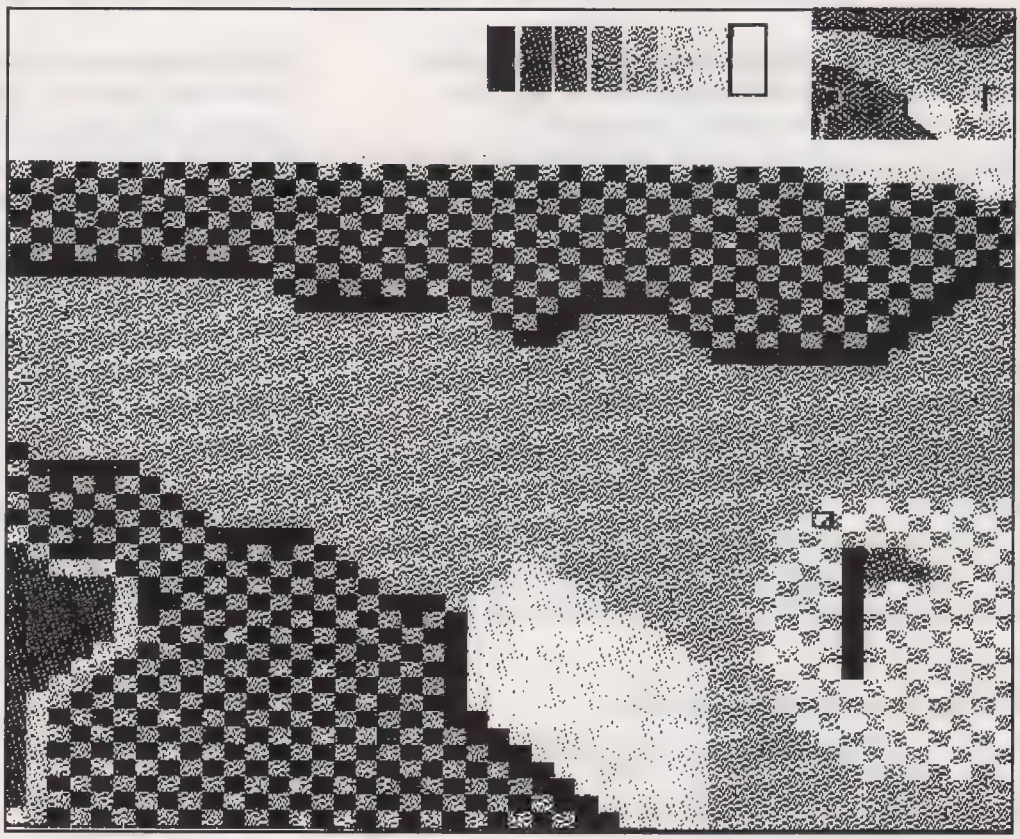
SOFTWARE FILE

THE PAINTER

INFORMATION

Program: *The Painter* V4.01, for QL with 256K+ expansion.
Price: £45 (+£10 bank costs) or 3000 Belgian Francs. Add 300 BFr if outside EEC. Cheques payable to Van der Auwera.
Publisher: Progs, Haachtstraat 92, B - 3020 Veltem, Belgium. Tel: (016) 48 89 52.

Rich Mellor has been attracted to the latest version of The Painter by its upgrades.



The Painter: Edit pixels one by one in Enlarge mode and see the effect immediately.

Throughout the QL's long life, there have been several drawing packages which varied a great deal in utility, speed and number of features. *The Painter* is the most recent in a long line up which includes such illustrious names as *GraphiQL Plus, MiceArt, M-Paint* and *Eye-Q*.

In 1989 I went to the Stafford Computer Show and saw The Painter. I was hooked after a few minutes using it, but decided not to purchase it at the time. Since then, several improvements have been made and when I received a special offer to buy a copy recently, I decided to take it up.

The current version is now fully *Minerva* compatible and has many improvements in speed compared to the earlier versions. There is a new manual in an A5 ring binder, which is more informative than previous manuals.

So why is The Painter special? On loading you soon notice the first difference. All the commands are chosen from menus on-screen. This option screen is actually separate from the main drawing screen – so that the screen does not become cluttered – and pressing 'ENTER' switches between the two screens. The number of different options seems rather daunting, but because the program uses Qjump's pointer interface, it is very simple to operate, and soon the manual becomes just something to refer to when using the more complex commands. There is a help function available from within the program which acts as a useful reminder on using the more complicated options.

The Painter appears to be a very professional program. It can be operated as easily from the keyboard or a joystick as from a mouse: a special configuration program allows you to specify either mouse settings or the speed at which the keyboard is read (if you do not possess a mouse). Users with full-size keyboards may experience a little difficulty in getting the settings so that the on-screen pointer does not move so fast as to be unusable, but with a little perseverance, the program can soon be set up to match your keyboard skills. I do not possess a mouse, but would love to see the program working with one, as the point trail mode (which leaves a trail of dots or a pattern on screen as

you move the pointer around) would be a joy to use.

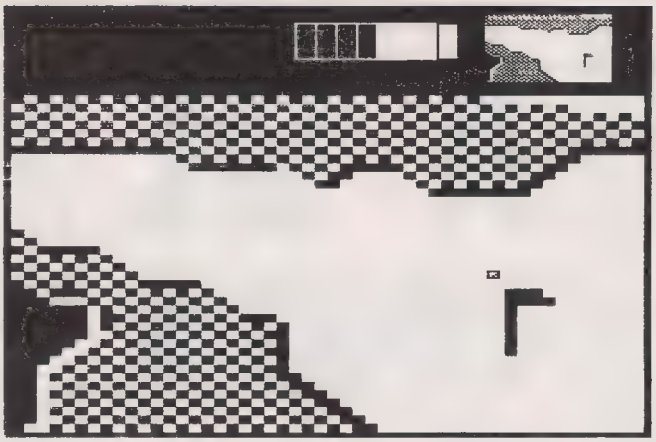
The program can deal with mode 4 and mode 8 screens (it can also convert them) as well as having all of the normal commands associated with drawing packages – circle, line, plot, square, fill, and many more. It does however also have some new and different commands for drawing pictures and for adding text to the screen.

On the drawing side, the standard QL ARC command has been replaced by a new 'Curve' command. This enables many more types of curves to be drawn. You simply choose the command from the menu screen, switch to the drawing screen (press ENTER), select the two ends of the curve and then use the cursor keys to move the high point of the curve.

There is also a 'Turn Shape' command which is actually more complex than its name may at first suggest. You need to outline the area to be 'turned' using a rubber-banded box. You can then specify four corners of any four sided shape into which the area will be moved. This can overlap the original, or be in a different place, but in any case, the transformation is smooth. It can even be used to re-shape a rectangle into a bow, for example. This command is very powerful, although it may also magnify/shrink the area which is to be re-shaped, and for this reason, I would have preferred the option to merely rotate a four sided shape as well.

Other graphics commands which are omitted by many of the other drawing packages include several methods of 'wiping' part of the picture, Pan/Scroll part of the screen (useful to line up parts of the picture exactly where you want them) and Zoom in/out (allows you to actually magnify part of the picture on screen if you decide that you drew something in the wrong size. All this takes place with a speed and ease of operation which leaves most programs well behind.

There is no need to worry if you make a mistake: if you press ESC after carrying out any of the commands, the screen immediately reverts to its previous state. Unfortunately this will only remove the last change to the picture, so for example, if you are drawing a series of lines, and decide to remove them altogether, you will have to use one of the Wipe options, which will also remove anything printed underneath the lines.



The same picture using the black and white screen jump option (here on a different print scale).

Colours on screen are easily chosen from a separate screen which allows you to either pick the colours and stipple manually, or from a large panel which shows all of the possible combinations. If neither of these suits your needs, the program can also use user-defined patterns which can be used to fill an area of the screen.

No mistake

When your drawing is nearly complete, the Screen Edit command comes into its own. This allows you to select a small area of the screen which is magnified so that it may be altered pixel by pixel. As you alter each pixel, a real-size representation of that part of the screen is shown in the top right corner so that you can see if the changes are having the desired effect. This is much quicker than the Magnified mode of GraphiQL Plus, although it does suffer a little by the fact that the pointer has to be moved to the top of the screen each time a different colour is needed. Unfortunately, this option cannot be used to line up the pointer on the overall screen, which would be useful, for example, to ensure that lines met each other.

Much thought has also gone into the text-manipulation commands, which allow text to be shown on screen in various ways. Not only can you use different fonts (standard QL fonts cannot be used, but there are several supplied with the program plus the option to design your own from within the program) and alter the character size, but you can also select to show the fonts in italics, outline only, bold, or even shaded (from four different angles).

There are two screens which allow you to specify how text is to be dealt with, the second of which allows even more complex operations. Text can be written not only sideways, but also on a slant and with a 3D effect (letters further along the slant appear smaller).

This text manipulation allows very complex screen displays to be created with writing in all possible directions, and in fonts up to 40x40 pixels in size (up to fifteen fonts may be loaded at any time).

The tools to help you produce a picture do not end here. The Painter can hold up to 12 different pictures at a time (depending on available memory), and parts of one picture can be lifted across to another screen using the filter and copy screen commands. The Filter command is useful if you set up a screen with several standard drawings on it, or just to doodle on. You could then opt to 'filter' out the background colour (eg black) which will enable this standard drawing to be placed anywhere on your current picture without affecting the background (useful for incorporating clipart).

Switching between the different screens couldn't be easier – you simply choose the View option, which then shows a small representation of all the open screens. You can then open or close a screen, move parts of one screen across to another, or simply opt to work on a different screen, all by moving the pointer using the cursor keys and pressing space. The multiscreen ability of The Painter must surely make it one of the most versatile drawing programs available for any home computer.

Once you have designed your screen, it can be saved to disk in normal 32K form or as a compressed screen. The compression routine is the best I have seen for the QL, although unfortunately at the moment there is no load-only version of the routine to allow people to use screens saved using it in their own programs.

If you want a hard-copy of the screen, an excellent screen-dump is included within the program. This can operate both 24-pin and 9-pin printers (it is apparently better with Epson compatibles), print the whole or part of a screen in three different sizes, print sideways, colour inverted, or black and white only (this turns off the shading used for the different colours).

There are many other options. Suffice to say that there are not many options which are not provided by this program (although I have yet to find a drawing package which can fill an area filled with a stipple colour).

There are no real problems with using the program (although some tv users may experience difficulty in reading some of the commands), however, despite it being written to run under Qjump's Pointer interface, oddly enough if you use a version of PTR_GEN later than that supplied with the program, you may end up with two pointers on screen, both moving about in response to the cursor keys!

As with any program, there are always one or two extras which the user adds to his/her wish list, but overall the program would seem to suit most people's drawing needs. It may seem a little expensive (mainly due to having to order it outside the UK), but I am certain that most users would say it was well worth it.

A QL COUSIN

The ICL One Per Desk is also a friend of David Warne's

The OPD (One Per Desk) computer was made by ICL Ltd. as a collaborative project between ICL, British Telecom and Sinclair with Psion providing the software. The same machine was badged for BT as the Merlin Tonto and for Australian Telecom as the Computerphone. The machine was intended for the busy executive with only limited computer skills. Most operations use multiple choice menus.

The hardware is based on the QL using the same 68008 processor, QL ulas, 128K of ram and microdrive data storage. The machine has a 'footprint' of about the same length but twice the depth of the QL.

Being launched shortly after the QL, the OPD suffered from the bad publicity attached to the microdrives, although ICL had much improved the reliability of the units. Despite this poor start, many hundreds were sold to local authorities, government departments and large companies. These are now coming on to the second hand market. They are regularly seen in magazines such as *Micro Mart* and *Computer Shopper*, with the cheapest units at about £100.00. Some computer supply companies are now replacing OPDs for their customers (mostly with PCs) and are selling the OPDs off cheaply.

A typical OPD featured a mono (black and white) monitor, twin microdrives, battery-backed clock, on-screen calculator, enhanced telephone handling, modem, Xchange software (*Abacus, Archive, Quill* and *Easel*), Basic and messaging (fax look-alike between OPDs).

The OPD is designed to be left on and the screen will blank if no keys are pressed for 10 minutes. Pressing any key, or an incoming call, re-activates the screen. The monitor is intended to be switched off between sessions leaving the computer powered for unattended functions. Monitors are available in 9 inch black and white or 14 inch Microvitec colour.

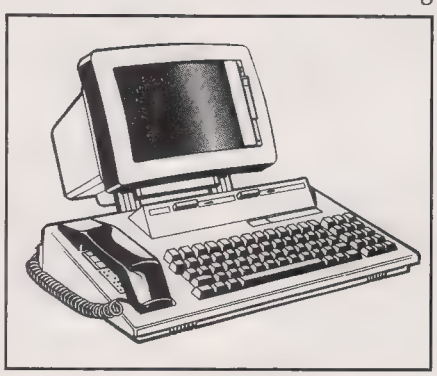
The microdrives are similar to those on the QL, but save the data in a different density. Although blank cartridges can be used on either machine, the OPD cannot read QL cartridge data. There is a program (for the QL) by Dave Walker of *DiscOver* fame, that can convert data and Basic from the OPD to the QL and vice versa. The OPD records cartridge use and read failures, and warns when the cartridge is due for renewal. The microdrives are very reliable.

The telephone has auto-dialing from a saved Telephone Directory/Address file of as many as 500 entries if required, with optional monitoring of cost and duration of calls. A running total is kept in memory. The directory has a search routine and short code dialling. There is a re-dial facility of any of the last six calls. Calls can be initiated through a built-in loudspeaker, the handset being picked up only when the connection is made. The machine will answer incoming calls using a programmable computer voice chip with different replies available for different times, eg lunch, holidays, gone home etc. There is, no facility to record incoming calls.

The modem is built-in and capable of Viewdata and Glass Teletype communications. This enables connection to Prestel, Yellow Pages, Tony Firshman's Board and many others. Screens can be saved to memory using the 'Snapshot' option, or entire programs can be downloaded to microdrive. Text can be prepared off-line to save phone charges.

The software is an enhanced suite of the programs supplied with the QL with the import/export of data between applications simplified. Being rom-based, it loads quickly and without read failures. The four are brought together under an operating 'shell' called Xchange Task Control. Up to eight 'tasks' can be in progress at one time. Import and Export between Psion programs is fast and simple. Xchange was an optional extra.

Basic is loaded from cartridge and is a reduced version of Superbasic. Many of the features of Superbasic are not available on the OPD. There are no graphics as such (CIRCLE, LINE, BORDER, FLASH etc.), no EXEC and no DIR. The screen size of the OPD is also slightly smaller. QL Superbasic programs can be transferred to the OPD but need considerable editing before use. Although using the same cpu, QL machine code programs cannot be run on the OPD. There are differences in the way the OPD handles the screen etc. that make QL programs incompatible.



Many OPDs were supplied with 'Messaging', a fax look-alike. This is in the form of a capsule that plugs into the back of the unit. The capsule contains a rom with the necessary code, which enables OPDs to send pre-typed text to each other using the telephone system. Received Messages can be edited and sent on to other users, printed or saved on microdrive. Later roms allow the messages to be sent at pre-set times and to different numbers to take advantage of cheap rate calls.

The OPD is provided with a serial port, but this works in one direction only, being intended solely for printer use. It is possible to download Basic files directly to the QL using the SER 2 port on the QL and a suitable cable. No input from the QL is possible by this route.

Later improvements included disk drives from PCML (with 256K extra memory), and another from Computer One, but these are no longer manufactured and can only be obtained on the second hand market. A variety of plug-in capsules were also provided, but most were to enable the OPD to connect to ICL mainframe computers and are of little use to enthusiasts. There were later options to allow the direct transfer of files direct from microdrive, via the telephone line, between OPDs and to import data into Quill or Abacus from Bulletin Boards.

It is not possible to simply plug-in extra memory as on the QL because the OPD requires special code to 'log-on and identify' the memory. A 128K expansion unit was made but few seem to have been sold.

Software is in very short supply, possibly because there is no organised user group such as Quanta, although some business oriented programs were produced. A diary/appointments program has been seen and also a CP/M operating system is available on one version of the disk drives. It is reported that Basic and 'C' compilers were produced but no sign of one has been seen.

QL owners who use their machine mostly for the Psion package will find the OPD easy to use and in a business environment, a very useful tool. Sadly, it suffers from the lack of compatibility with the QL in the most important area, programs. Very little software is produced for the QL in Superbasic now and useful programs like *Flashback* and Tony Tebby's utilities cannot be used. The most promising area lies in Xchange applications and it is possible to transfer the *Archiver* group of programs once sold by Eidersoft.

I have both an OPD and a QL and swop data between them regularly. I like both of them. The OPD has made me friends in Scotland, Eire and Australia and the QL many at the local Quanta group.

SINCLAIR QL WORLD

THE NEW USER GUIDE

Part 8 of the New User Guide turns to handling data in numeric and character arrays, and setting up multi-dimensional arrays. Channels are used to send an array to mdv and retrieve it again.

SECTION EIGHT

Data

So far in our exploration of SuperBasic we have been concentrating on the mechanics of getting the computer to do what we want it to. The language's procedures, functions and control structures have been examined and put to some use, but a very important omission must now be put right. The most valuable part of a computer is the data it contains, yet in our efforts to understand SuperBasic, the data it is designed to manipulate has taken a back seat.

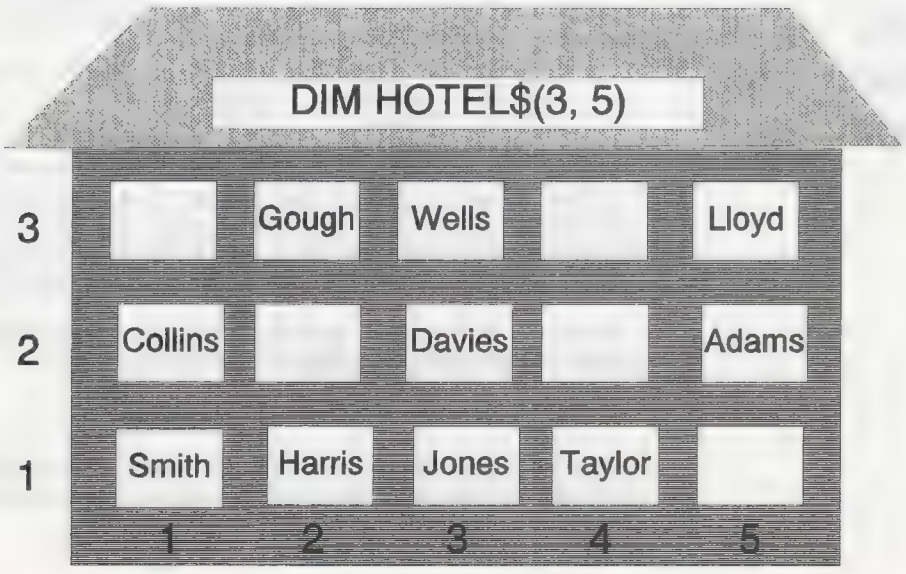
Even when it is first turned on, the QL holds lots of information. It knows SuperBasic and Qdos, microdrives, the alphabet, simple mathematics, trigonometry and much else. These things are knowledge, but they are not data. In computing, 'data' has a narrow meaning, referring to information specific to a computer application. The banking details in an accounts program, the customer details in a marketing database, the possible moves in a chess program and the page objects in a desktop publishing package are all data.

Incidentally, there is some confusion over whether 'data' should be singular or plural. Strictly, a single piece of information is a datum and several pieces are data, but it is normal to treat 'data' as a group noun. 'Datum', which is a Latin word meaning "a thing which is given", is rapidly becoming an archaic word which is little used in English speech. 'Database' is on the other hand a relatively new word used to refer to any organised collection of information, but which is especially used to describe information held on a computer in an organised and accessible form.

What little data has featured in the *New User Guide* up to now has been inconsequential and transient. No sooner was it created than it was printed on the screen and forgotten. Data is not like that in real life. Data tends to be bulky, organised into sets and, usually, valuable enough to keep. SuperBasic needs to be able to deal with such data if it is to be a genuinely useful programming language.

Arrays

In Basic, data is held in variables, but the variables we have used up to now have held only one piece of information. For instance, the computer could hold a single customer's name in a variable called Customer$, but this would be of little value if the details of hundreds of customers had to be stored and manipulated. SuperBasic handles multiple pieces of similar information in data structures called arrays.



At its simplest, an array can be imagined to be like a table in a reference book, for instance:

CUSTOMERS
J Smith
M Jones
S Taylor
A Davies
L Collins

DIMension

To create such an array in SuperBasic a special command is used to define the array and ensure that sufficient memory is set aside to hold the array's contents. This command is DIM, which is short for 'dimension', and it is followed by the name of the array with its size in brackets. Array names follow the usual rules for naming variables, including the rule that requires a $ sign at the end of the names of arrays and variables which hold text data. The command:

```
100 DIM Customer$ (100,30)
```

creates an array called Customer$ which has room for 100 entries, each of which can have up to 30 characters. SuperBasic is an exception to most Basic dialects in that it requires the programmer to state the maximum length of each element in a string array – most Basics delay allocating memory space until a string array element is filled and so have no need for a maximum length.

In this respect, SuperBasic is more modern, like the C programming language and most current database languages. In SuperBasic it is possible to define an array comprising a single string, eg DIM Example$ (40), meaning a string with a maximum of 40 characters. This becomes very useful when a compiler such as Digital Precision's *Turbo* is used, and it will also make C's string handling familiar should your programming requirements ever take you in that direction.

Numeric arrays are even easier than character arrays. If each customer's account balance is to be held alongside their names, a suitable array could be defined with the command:

```
110 DIM Balance (100)
```

With numeric arrays it is not necessary to declare how long the longest number will be because SuperBasic holds all numbers from zero upwards in the same amount of memory space.

'Subscripts'

Information is placed into and read from arrays using a subscript to determine which array element is involved. If the twelfth customer is M Larkin with an account balance of £50, this information could be stored with the following commands:

```
120 Customer$ (12) = "M Larkin"
130 Balance (12) = 50
```

The idea of subscripts is very valuable because of the flexibility it gives to programmers. For example, here is a program fragment which prints out the customer database with the appropriate account balances:

```
200 FOR x = 1 TO 100
210     PRINT Customer$(x),,Balance(x)
220 END FOR x
```

It is perfectly possible to make valuable use of arrays without straying far from the very basic principles so far covered, but the adventurous can take arrays much further. The two arrays used above are 'single dimensioned arrays', in other words they both represent a single column of data.

Multi-dimensional arrays

SuperBasic supports multiple-dimensioned arrays. A two-dimensioned array is like a table with many columns and rows, such as:

J JONES	10 HIGH STREET	WORCESTER
T SMITH	15 THE AVENUE	YORK
A HARRIS	22 ACACIA AVENUE	GLASGOW

A three-dimensioned array might be used to refer to a set of such tables in a book, while a four-dimensioned array might refer to sets of tables in books in a number of libraries. Arrays with more dimensions are catered for quite happily in SuperBasic but programmers tend to have difficulty imagining them, so let us stick with two-dimensioned arrays for now.

Imagine a hotel which has six floors each with twenty bedrooms. The computer can hold the names of each bedroom's occupant in an array defined as:

```
100 DIM Room$ (6, 20, 10)
```

The array represents a matrix of 6 rows and 20 columns and the content of each matrix cell is limited to 10 characters. We can now write procedures which handle several likely occurrences, such as:

LISTING 1

```
100 REMark : Example arrays
110 :
120 REMark : Simple numeric array
130          DIM balance (40)
140 :
150 REMark : Simple string array
160          DIM customer$ (40, 20)
170 :
180 REMark : Multi-dim numeric array
190          DIM demo (10, 10, 20)
```

```
200 REMark : Fill two arrays with dummy data
210 FOR x = 1 TO 40
220   balance(x) = RND(100)
230   customer$(x) = "Mr Smith"
240 END FOR x
```

```
250 REMark : Save both arrays to floppy disk
260 StoreBalance  : REMark see Listing 2
270 StoreCustomer : REMark see Listing 2
```

```
280 REMark : Re-initialise both arrays
290 DIM balance (40)
300 DIM customer$ (40, 20)
```

```
310 REMark : Now restore data from floppy disk
320 FetchBalance  : REMark see Listing 3
330 FetchCustomer : REMark see Listing 3
```

```
340 REMark : Prove that arrays are populated
350 PRINT balance
360 PRINT customer$
```

Listing 2

```
500 DEFine PROCedure StoreBalance
510   DELETE flp1_balance
520   OPEN_NEW#3, flp1_balance
530   PRINT#3, balance
540   CLOSE#3
550 END DEFine
```

```
600 DEFine PROCedure StoreCustomer
610   DELETE flp1_customer
620   OPEN_NEW#3, flp1_customer
630   PRINT#3, customer$
640   CLOSE#3
650 END DEFine
```

Listing 3

```
700 DEFine PROCedure FetchBalance
710   OPEN_IN#4, flp1_balance
720   INPUT#4, junk
730   REMark: removes the 0 element
740   FOR x = 1 TO 40
750     INPUT#4, balance (x)
760   END FOR x
770   CLOSE#4
780 END DEFine
```

```
800 DEFine PROCedure FetchCustomer
810   OPEN_IN#4, flp1_customer
820   FOR x = 1 TO 40
830     INPUT#4, Customer$(x)
840   END FOR x
850   CLOSE#4
860 END DEFine
```

Book Mr Jones into Room 3 on the fifth floor:

```
Room$(3, 5) = "Mr Jones"
```

Find out who is in Room 12 on the second floor:

```
PRINT Room$(2, 12)
```

Record that Mrs Davies has left Room 15 on the first floor:

```
Room$(1, 15) = ""
```

Restrictions on arrays

Unfortunately, SuperBasic places two restrictions on the use of multi-dimensioned arrays which make them less helpful than they might otherwise be. Firstly, the array must contain all strings or all numbers, whereas other languages allow an array to have a mix of strings and numbers just as a reference table might have text and numeric columns. Secondly, SuperBasic insists that the cells in string arrays are all the same size, so you cannot save space by declaring some short elements and some long elements.

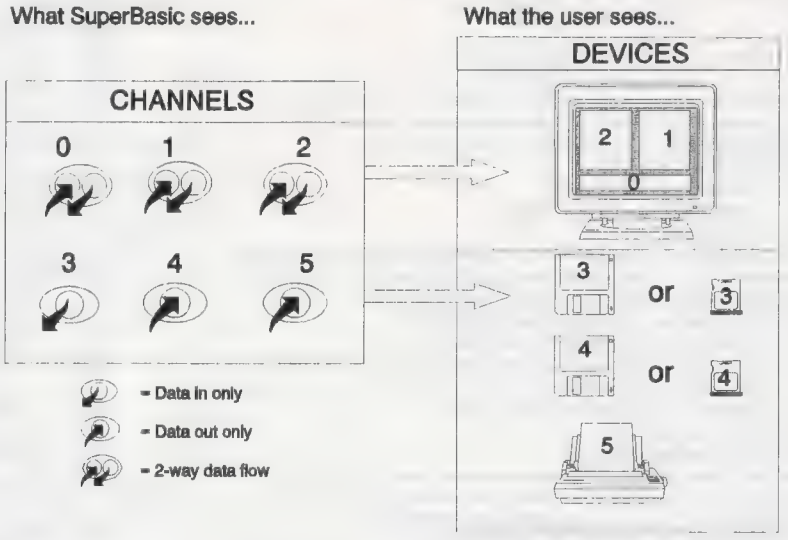
The only way round these problems is to use lots of arrays and make sure that the subscript numbers match as they did in the earlier example of Customer$ and Balance. If you are designing a database, some thought needs to go into the way arrays will be used. To extend the hotel example a little, arrays might be needed to store the dates of arrival and departure of each guest, the morning newspaper they order, their bill total, and so on. The arrays might look like this:

```
100 DIM Room$        (6, 20, 10)
110 DIM Arrival      (6, 20)
120 DIM Departure    (6,20)
130 DIM Newspaper$   (6, 20, 14)
140 DIM Bill         (6, 20)
```

Memory use

It is possible to allocate a great deal of memory space using arrays like this, so a rough calculation is necessary to ensure that the arrays will fit into the QL along with the program necessary to access the data. Numbers occupy five bytes no matter what size the number actually is on screen, so for each of the numeric arrays a total of 6*20*5 bytes will be used, or 600 bytes. The string arrays will occupy 6*20*10 = 1200 bytes and 6*20*14 = 1680 bytes.

The raw data in this small database will eat up no less than 4680 bytes of valuable memory space. Arrays also need additional memory space to hold 'pointers' which allow the computer to find its way around the array matrix. In a string array each element has an overhead of two bytes and each dimension has a further overhead of two bytes. Numeric arrays only need two bytes per dimension, which is usually a negligible amount. Before writing your magnum opus home accounts program it is a very good idea to calculate exactly how much memory space will be needed to store the payee, amount, description and date for every cheque you might write in a year.



Storing an array via a channel

Part 5 of the New User Guide broached the subject of channels. Channels can be used to connect the QL with any device, such as printers, modems, microdrives and disk drives. Once the connection is made data can be sent to the device or received from it via the channel. To conclude this section on data arrays we shall use channels to save an array to a microdrive file and to retrieve it back into memory. These are important techniques because a great deal of time can be expended filling arrays with information which users wish to view again at a later date.

Sending an entire array to a file could not be easier because SuperBasic allows the whole array to be copied with a single command. A file is opened and linked to a channel, the array is copied using the PRINT command, and the file is closed again. **Listing two** shows how this is achieved.

Within the file the array loses much of its structure. Each element is recorded separated by a line feed, so if the file is viewed on the screen using the command:

```
COPY MDV1_ARRAYFILE TO SCR_
```

the display will list all the elements in order. The file will not contain information about the array's name, its total size, the length of its strings or the number of dimensions it has.

Restrictions on arrays

Retrieving the array contents is, accordingly, a little more difficult. Firstly, if the original array no longer exists (because perhaps the program using the array has been re-loaded) the program must declare the array using the correct name and details. The elements can then be inserted from the storage file using a loop for each array dimension. The examples at **Listing three** relate to single-dimensioned arrays and are therefore quite straightforward.

An interesting point to note is that numeric arrays have a 0 element, so an array declared by DIM number(9) will have ten elements with subscripts from 0 to 9. Because string arrays do not have a 0 element it is customary to ignore the existence of the 0 subscript in numeric arrays. In Listing three the 0 element is read from the file and ignored.

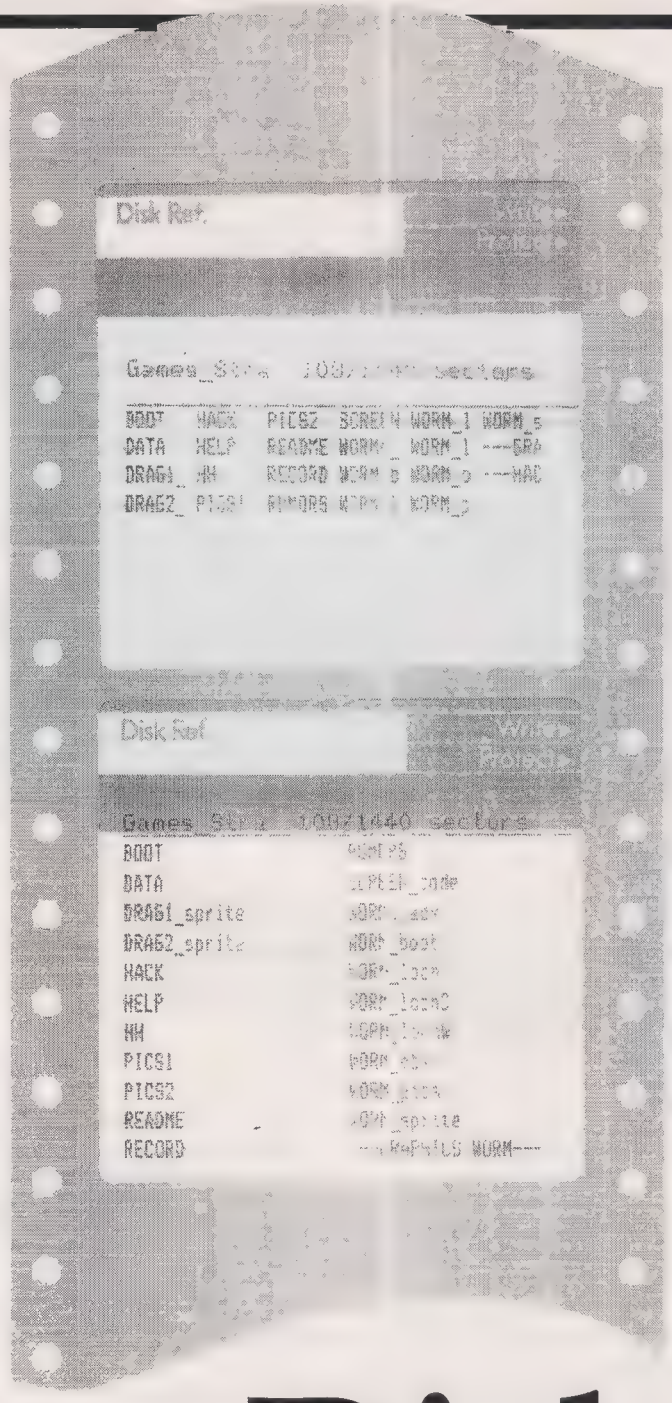
With any database program there is a fairly well-defined set of activities which need to be carried out to make the data useful. The program must be capable of adding, altering, displaying and deleting data. It must be able to store data between program sessions and restore the data at the start of a session. The program must usually be able to sort data into order and group items according to a variety of criteria. The program will also need to be able to print the database, or selected parts of it.

All these things are possible within SuperBasic, although the limitations of the Basic dialect can make some of the operations quite slow when dealing with large amounts of information. Most of the activities require loops to be set up. For example, if someone's cheque records were held on the computer it would be possible to construct a loop written over a given period. Sort routines can be highly complex in their use of loops: *Sinclair QL World* has published dozens of SuperBasic sorting routines in its time.

Plan ahead!

Before contemplating a major database program, it is advisable to spend a great deal of time planning before programming commences. Detailed advice on program planning has been published in the *Better Basic* and *Super Basic* series. It is also worthwhile to consider the merits of using a SuperBasic compiler such as *Q-Liberator* or *Turbo*.

INFORMATION
Program: *Super Disk Labeller*
Supplier: Dilwyn Jones Computing, 41 Bro Emrys, Tal-y-bont, Bangor, Gwynedd. Tel: 0248 354023.
Price: £10.00 (p&p included) for 3.5 and 5.25 disks or mdvs. 256K memory required.



Super Disk Labeller

John Shaw delves into the possibilities offered by a small package dedicated to sorting and labelling disk files.

Here we are then and just when I needed it. the program to make tidy printed labels for your floppy disks and what a good job it does. I have always wanted my Quanta library disks to have the contents listed on their labels and now I can do it.

This program can print labels that show the filenames on the disk in small print in several columns, plus a heading or title. It can also optionally include statistics such as file size, type and update date.

The reconfigurable printer driver with the package gives you control over the size of the text used and hence the amount of filenames that can be catalogued on the label.

Sorted

Filenames can be sorted into alphabetical order and even grouped so that, for example, only Quill_doc files are listed. Once the label has been assembled you can then edit the text, if required, or add your own text if there is room. There are quite a number of permutations possible.

There are also three special facilities.

Number one is the Quick Label facility which simply makes a sorted multicolumn filename label for the disk in a default drive with the minimum of key presses from the user.

Number two is a Sleeve Insert which is a list of filenames on a piece of paper larger than a label for keeping with the disk, especially if you use sleeves with your disks – mostly relevant to 5.25 in disks, although some 3.5 in disks are also supplied with clear plastic sleeves. With this facility, you can print a sheet showing filenames on a disk together with the disk name, which can be filed away easily.

Printer

Before using the program, you should select the printer driver to use with your printer. Several printer drivers are supplied with the program as follows:

DEFAULT_DRIVER: This is a very simple driver using very basic control codes that should work on just about any printer.

9 PIN_DRIVER This is a general purpose printer driver that seems to work with all Epson compatible printers that can print in condensed elite subscript print. It has been tried before release on Star LC10 and Centronics GLPII printers quite successfully.

9 PIN_PROP_DRIVER This is a proportional spacing printing version of the above.

24 PIN_DRIVER A 24 pin version of the nine pin driver above.

24 PIN _PROP_DRIVER A proportional spacing version of the 24 pin printer driver.

The program is driven by function key menus: F1 for select drive and read directory, F2 for sort, group and select filenames, F3 for make/view/edit label and backup, F4 for printouts and printer drivers and F5 for quick label. It is capable of reading what is on a drive, assembling a label and starting to print it after it has

sorted the filenames into order.

The printing will consist of a heading, made up of the disk title and capacity in sectors. All columns are the same size and the width of the column depends on the longest filename found on the disk.

The Sleeve insert facility enables you to print a piece of paper that can be stored with the disk in its sleeve, if you use them, or can be used in a disk box to remind you where each disk goes.

Selected

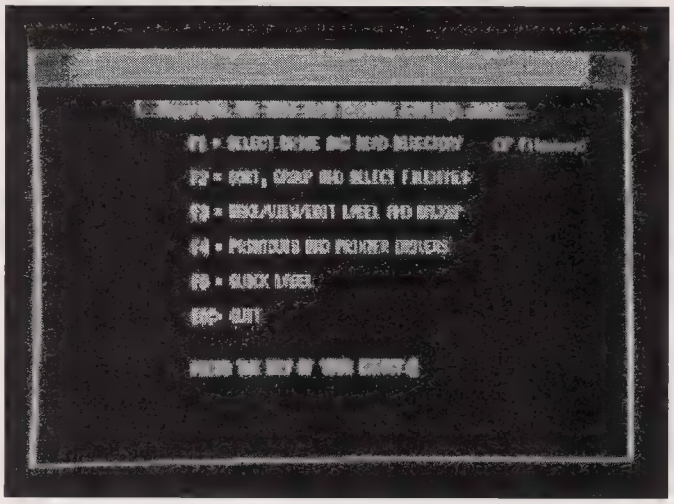
The Files Tidy List is a complete list of selected files on a disk, printed on a normal sheet of paper. It differs slightly in that it allows you to manipulate the list of files on a disk first, such as selecting only the _doc files, for example. The purpose is for you to be able to store on paper, for easy reference, a master list of files that you can look up at a later date. As it allows you to print on whole pages, you can use it for disks that have a very large number of files.

The Main Label making function is a little more complicated than the Quick Label function. To enable you to perform more manipulations on the label content before printing it, the filenames are read into an array list, where you can sort them into order, group by prefix (filenames starting with given letters or directory) and suffix (filenames with a certain extension only, such as _doc files).

Menu

On the same menu there are other processes that can be applied to the list. The first and most obvious is to sort it into alphabetical order. You can opt to group a list of files according to a suffix or extension name. For example, if you wanted all the _doc files listed together and all the _dbf files listed together. Up to ten extensions can be listed for the program to do the matching.

The program runs through the list of filenames, placing them in order as it finds the matching extensions. On the Super Disk Labeller disk, for example, you could list all the printer driver files together by using the extension _dat.



"This program can print labels that show the filenames on the disk in columns and optionally include file sizes, etc."

There is also a similar facility to group filenames by their root. Root in this case means the first few characters of a name, which could simply be a letter or two at the start of a name.

The final option on this menu allows you to undo any sorting or grouping operations that you may have performed on the files list. It simply restores the original list as it was read from the disk by killing the index list within the program.

Once you have sorted, grouped and selected all the files you want into the order you want, you then have to make a label. This is a copy in memory of what the label will be like when printed. After building up the label layout you can then preview and edit the lines of text.

The first two fixed types are the commonly available sizes of labels on a roll for 3.5 in disks. Type one is the larger of the two types. This is the type of label that folds round the edge of the disk. The second type is a label that does not wrap around the edge, but fills the label area on a 3.5 in disk.

Others

The 5.25 in labels are the third type. These are smaller labels that fit in one corner of the disk. This type will probably not be used much as most QL owners use 3.5 in disks. The fourth type is an address label, 3.5 inches wide and 1.5 inches deep. The final type calls on you to enter the label size yourself.

Next, you are asked for the text area layout details. Five choices are shown on the screen: single column, filenames only, single column, with statistics, multiple columns, with full filenames, multiple columns, best fit and squashed best fit. Once you have sorted, grouped and selected the list of files as you want to, there is one more special facility. The Backup utility is a short method of making a backup copy of the disk. The files are copied in the order held in the list in Super Disk Labeller. It can look very professional when all the filenames are listed in alphabetical order or grouped when looking at the disk with the DIR command in Basic.

There is a Customise Driver option that lets you edit existing printer drivers or create new ones specifically for your printer. There are some thirty sets of codes to enter, so you will need your typing fingers in good working order and plenty of patience.

Options

To give you some idea of the available options, a selection from the menus include: Group by Root, Group by Extension, Manual selection, Sleeve insert printout driver, Files tidy list printout, Select label types, Backup selected files, Customise printer, Use default printer.

So, here we have a superb piece of programming utility, containing all the possible options you could wish for and sold together with a 20-page User Manual which I found easy to use and follow.

In addition, if you have a very unusual printer, you can contact Dilwyn Jones and he will do his utmost to be of assistance.

A good, well thought out package, representing excellent value for money.

```
Games Stra  108/1440 sectors  (23 filenames)
BOOT                HACK                PICS2               SCREEN_code         WORM_locnC          WORM_sprite
DATA                HELP                README              WORMc_adv           WORM_locnW          ---GRAPHICS WORM---
DRAG1_sprite        HH                  RECORD              WORM_boot           WORM_obj            ---HACK---
DRAG2_sprite        PICS1               RUMORS              WORM_locn           WORM_pics
```

AMIGA

STAND-ALONE 500 COMPUTER

● 512K AMIGA	£329.99
● TV MODULATOR	FREE
● MOUSE CONTROLLER	FREE

***PLUS!* - FREE FROM SILICA**

● ARCADE ACTION PACK	£229.78
● PHOTON PAINT 2.0	£89.95
TOTAL RRP:	£649.72
YOU SAVE:	£320.72
SILICA PRICE:	**£329**

PLUS! ***FREE*** HOLIDAY ACCOMMODATION

£329

SILICA PRICE - INCLUDES VAT + FREE DELIVERY

NEW! CARTOON CLASSICS PACK

HARDWARE:

512K AMIGA 500	£329.99
A520 TV MODULATOR	FREE
A501 RAM EXPANSION	£99.99

SOFTWARE - ENTERTAINMENT

BART SIMPSON	£24.99
CAPT PLANET & PLANETEERS	£25.99
LEMMINGS	£25.99

SOFTWARE - PRODUCTIVITY

DELUXE PAINT III	£79.99

PLUS! FREE FROM SILICA

ARCADE ACTION PACK	£229.78
PHOTON PAINT 2.0	£89.95
TOTAL RRP:	£906.67
YOU SAVE:	£507.67

PLUS! 16 NIGHTS FREE HOLIDAY ACCOMMODATION

£399

SILICA PRICE - INC VAT + FREE DELIVERY

FREE! FROM SILICA WITH EVERY AMIGA

ARCADE ACTION PACK

The new Arcade Action Games Pack is THE software compendium for Amiga owners, featuring ten top titles with a variety of different types of games for you to enjoy. Each title is packaged in its own plastic case, with a colour sleeve and full instructions. The Arcade Action Pack is FREE! when you buy your Amiga 500 from Silica.

FREE! WORTH: **£219.78**

FREE! PHOTON PAINT 2.0 GRAPHICS PACKAGE

Every Amiga 500 from Silica Systems comes with a FREE Photon Paint 2.0 (RRP £89.95), one of the most advanced art packages available for the Amiga. With it, you will be able to harness the extraordinary graphics power of the Amiga and produce inspirational pictures in minutes. With 4096 colours, your pictures will reach life-like appearance. These can be animated in real-time using a vast range of graphical effects, including blending colours and dithering, stencils, shadowing, contour mapping and surface mapping.

WORTH **£89.95**

FREE! 16 NIGHTS HOLIDAY HOTEL ACCOMMODATION

Every Amiga from Silica comes with a FREE 72 page, full colour brochure with accommodation vouchers. These vouchers entitle 2 people to stay up to 16 nights in one hotel (or any number of hotels to a total of 16 nights) with accommodation FREE (you could take up to six nights for four people, or other options). Choose from the 250 hotels featured in the catalogue. All you have to pay for are your meals (prices are listed in the brochure). Reduced rates in top London hotels and hotels in France are also featured.

TOTAL FREE PACKAGE

***ARCADE ACTION PACK* - £229.78:**

Asterix - *By Coktel Vision*	£24.99
Chess Player 2150 - *By CP Software*	£24.95
Drivin' Force - *By Digital Magic*	£24.95
Live & Let Die - *By Elite*	£19.99
Onslaught - *By Hewson*	£24.99
Pipe Mania - *By Empire Software*	£24.99
Rick Dangerous - *By Firebird*	£24.99
Rock 'n' Roll - *By Rainbow Arts*	£19.99
Skweek - *By US Gold*	£19.99
Trivial Pursuit - *By Domark*	£19.95

***ART PACKAGE* - £89.95:**

Photon Paint 2.0 - *By MicroIllusions*	£89.95
TOTAL RRP:	£319.73

FREE! WORTH NEARLY **£320**

PLUS! ***FREE*** HOLIDAY ACCOMMODATION

SILICA SYSTEMS OFFER YOU

- ***FREE OVERNIGHT COURIER DELIVERY:*** On all hardware orders shipped in the UK.
- ***TECHNICAL SUPPORT HELPLINE:*** Team of Amiga technical experts at your service.
- ***PRICE MATCH:*** We normally match competitors on a "Same product - Same price" basis.
- ***ESTABLISHED 12 YEARS:*** Proven track record in professional computer sales.
- ***BUSINESS + EDUCATION + GOVERNMENT:*** Volume discounts available for large orders.
- ***SHOWROOMS:*** Demonstration and training facilities at our London & Sidcup branches.
- ***THE FULL STOCK RANGE:*** All of your Amiga requirements from one supplier.
- ***FREE CATALOGUES:*** Will be mailed to you with offers and software/peripheral details.
- ***PAYMENT:*** By cash, cheque and all major credit cards.

Before you decide when to buy your new Amiga computer, we suggest you think very carefully about WHERE you buy it. Consider what it will be like a few months after buying your Amiga, when you may require additional peripherals or software, or help and advice with your new purchase. And, will the company you buy from contact you with details of new products? At Silica Systems, we ensure that you will have nothing to worry about. We have been established for over 12 years and, with our unrivalled experience and expertise, we can now claim to meet our customers requirements with an understanding which is second to none. But don't just take our word for it. Complete and return the coupon now for our latest Free literature and begin to experience the "Silica Systems Service".

MAIL ORDER HOTLINE 081-309 1111

SILICA SYSTEMS

MAIL ORDER:	1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX			**Tel: 081-309 1111**
Order Lines Open: Mon-Sat 9.00am-6.00pm	No Late Night Opening			Fax No: 081-308 0608
LONDON SHOP:	52 Tottenham Court Road, London, W1P 0BA			**Tel: 071-580 4000**
Opening Hours: Mon-Sat 9.30am-6.00pm	Late Night: Thursday until 8pm			Fax No: 071-323 4737
LONDON (SELFRIDGES):	1st Floor, 369 Oxford Street, London, W1A 1AB			**Tel: 071-629 1234**
Opening Hours: Mon-Sat 9.30am-6.00pm	Late Night: Thursday until 8pm			Ask for extension 3914
SIDCUP SHOP:	1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX			**Tel: 081-302 8811**
Opening Hours: Mon-Sat 9.00am-5.30pm	Late Night: Friday until 7pm			Fax No: 081-309 0017

To: Silica Systems Ltd, Dept. QLW 1091-60, 1-4 The Mews, Hatherley Rd, Sidcup Kent, DA14 4DX

PLEASE SEND FREE AMIGA COLOUR CATALOGUE

Mr/Mrs/Ms: ..... Initials: ..... Surname: .....

Address: .....

.....

.....

Postcode: ..... Tel: .....

Systemati
Machine Code
Programm

The first in a new series by Alan Bridewell, building on basic Machine Code skills.

I should begin by asking the questions: 'Why bother with machine code? Why not stick to Basic, or in the case of the lucky QL owner, stick to SuperBasic, surely the best implementation of the language on any machine?'

Up to a point, SuperBasic is fine. I write SuperBasic programs, and use other people's SuperBasic programs all the time, with no complaints. None, that is, until I need some rapid screen handling, or I need them to multi-task, or I'm sick of waiting forever for the program to load, or . . .

The problem is, the microprocessor in the QL (or any other computer) doesn't understand SuperBasic. So each command has to go through the interpreter, which converts it into numbers the microprocessor can understand. As a result, the QL spends most of its time trying to work out what the commands mean, and only a tiny fraction of its time actually carrying them out. Even worse, the microprocessor *never* learns, so no matter how many times your SuperBasic program loops through the same set of commands, they *always* need interpreting over and over again. What a waste of time!

The next question should be: 'What about compiled SuperBasic, using *Supercharge, QLiberator,* or *Turbo*?'

Compiler

A SuperBasic compiler will turn your SuperBasic program into a set of instructions the microprocessor can understand, so it doesn't have to wait for the interpreter, and can carry them out at once. It's magic to see your sluggish SuperBasic program become a fast, compiled program. And not only that, but it multi-tasks as well. What more could you want?

Well, there are ways, and ways, of telling a microprocessor what SuperBasic was designed to tell it to do. Compilers tend to be long winded. As a result, the program tends to get very long, and expensive in ram use. Not very useful if you want several multi-tasking in your limited ram space. Not only that, but often there are much faster sets of instructions than the compiler can find. So although compiled SuperBasic is much faster than interpreted SuperBasic, the program could be much faster still.

Which leads us to the question: 'What about other compiled languages?'

SuperBasic was designed to make things as easy as possible for human beings to tell the QL what to do. As a result, we get all the problems so far mentioned. Other languages, like Forth and C, are more of a half-way house. They are rather more difficult for us humans, but not *that* difficult, so most of us can pick them up if we're prepared to put the time and effort into learning them. The big advantage they have over compiled SuperBasic is that the compiler converts them into instructions that are more compact, and faster. But, even so, none of them can do it perfectly. The microprocessor can still be instructed in a more concise, and faster, way. But, in the end, the only way to do this is to instruct the microprocessor in its own language.

Unfortunately, the microprocessor's own language is an ever-changing series of numbers, which only computing masochists would attempt to use. But help is at hand, in the form of *assembler language.* This consists of a set of instructions which human beings can easily understand, but where each instruction converts to a unique number (or set of numbers) in the microprocessor's machine code language. For instance, in assembler language, we could write.

```
MOVE. L D4, D2
```

which simply means 'Copy the long word stored in the D4 register of the microprocessor into the D2 register of the microprocessor'. The assembler will convert this into the hexadecimal number $2404, which the microprocessor will recognise as that precise instruction, and writes it into the program. What could be easier? If some experts are to be believed, nothing is easier.

Well, in spite of what some of the experts say, it is a fact that the vast majority of computer users find machine-code programming very difficult, if not actually impossible. So, why should this be?

It's not that following the architecture of the microprocessor is difficult. Ideas like address, data, register and program counter are easy enough. And the lines of program are easy enough in themselves. Some of the instructions are quite subtle, but with a bit of thought they are all understandable.) With some effort, most of us who dabble in machine code can follow, say, Simon Goodwin's *DIY Toolkit* routines, and possibly make slight modifications to the code to suit our own needs. But to go from there to writing whole programs from scratch is not on.

The problems, I believe, come down to these.

1. It is too big a jump of imagination to go from single program lines to working out what large chunks of code will do. It's one thing to write code which simply moves numbers around registers and ram. It's quite another thing to work out how doing this can make things appear on the screen, or open files, or print to the printer, or any of the real jobs you want your program to do. We need small chunks of code which perform a recognisable task.
2. Many (possibly all) of these chunks of code are already there as trap calls or vectored routines. However to use them we have to put certain data in certain registers, and then sometimes look at other registers to find the result of calling the code. As it's almost impossible to remember where all this data has to go, it means continually referring to a m/c programming manual every couple of lines we write.

As a self-taught programmer, who does it for amusement, I have worked out a system which has enabled me to produce a number of useful programs.

What I have produced, and what I would like to share with you, is a programming kit consisting of small chunks of code to perform particular functions. They are fully annotated to show how they work, and how they may be fitted together with other chunks of code. In particular, special note is made of any alterations they might need

c
ing

in particular circumstances. For instance, every time the stack is used, the effect on the stack pointer is noted, so that when we need to use some data stored on the stack, we know where it is stored in relation to the current stack pointer. Also, if the code requires any data blocks, or buffers, or equates, these are set out so they can be conveniently moved to the end of the code to be lumped together.

At this point I should say that I program using *Assembler Workbench* by Talent. (You should have no problems modifying the listings to suit your assembler.) I learned most of my machine code work by reading Adam Denning's *Advanced QL Machine Code,* and a lot of what I have was produced by taking his programs apart and using the bits. I have also referred much to Adrian Dickens' *QL Advanced User Guide,* and, of course, to the many m/c programs that have been published over the years by *QL World,* and before that by *QL User.*

I should also point out that these articles are not designed as a beginner's tutorial in assembler language. (If you are not familiar with it you should obtain a suitable guide.) These articles are aimed at helping the many who can read programs to be able to write programs.

We will start by listing some of these chunks of code, and, where necessary, explaining what they do and how they do it. Then later, we shall put some of them together to make a program.

Jobstart (Listing one)

'Jobstart' is the 'standard' way to start the code for a multi-tasking job. I'm not sure it's actually necessary, but it does allow you to identify the job among others, using a toolkit command which prints out the state of the current jobs in the machine. (I don't know how it works - perhaps someday I'll find out.)

Priority (Listing two)

Often the first thing we want a job to do is to set its priority to fix what proportion of the processor's time it is going to use. (Basic has a priority of 32.)

'Priority' is a TRAP 1 call. We must put #MT_PRIOR in DO (#MT_PRIOR is the hex number B). We must put the job ID in

Listing 1

```
; ************************************************************************
;                              'JOBSTART'
; ************************************************************************
; THIS SHOULD BE HOW A MULTI-TASKING JOB SHOULD BE STARTED.
; START IS THE ADDRESS OF THE FIRST INSTRUCTION OF THE JOB.
; THE CHARACTER COUNT AND JOB NAME SHOULD BE EDITED AS APPROPRIATE.
;
                    BRA.S     START       ; BRANCH TO START OF CODE
                    DC.L      0           ; (THIS IS STANDARD FORMAT FOR
                    DC.W      $4AFB       ; START OF A JOB)
                    DC.W      4           ; CHARACTER COUNT OF JOB NAME
                    DC.B      'TEST'      ; NAME OF JOB
;
; START SHOULD NORMALLY POINT TO CODE SETTING THE PRIORITY OF THE JOB
; ************************************************************************
```

Listing 2

```
; ************************************************************************
;                              'PRIORITY'
; ************************************************************************
; THIS CODE SETS THE PRIORITY OF THE MULTI-TASKING JOB.
; THE VALUE IN D2 SHOULD BE ADJUSTED TO THE PRIORITY REQUIRED
;
.PRIORITY           MOVEQ     #$B,D0              ; #MT_PRIOR IN D0
                    MOVEQ     #-1,D1              ; OF THIS JOB
                    MOVEQ     #1,D2               ; TO 1
                    TRAP      #1
;
; ************************************************************************
```

Listing 3

```
; ************************************************************************
;                              'CONSOLE'
; ************************************************************************
; THIS IS HOW WE SET UP A CONSOLE WINDOW.
; MOST JOBS WILL REQUIRE AT LEAST ONE CONSOLE WINDOW.
;
; OPEN THE CONSOLE CHANNEL
;
                    LEA.L     PBLOCK,A1           ; PBLOCK ADDRESS IN A1
                    MOVE.W    $C6,A2              ; UT_CON VECTOR IN A2
                    JSR       (A2)
;
; SAVE THE CHANNEL ID WHICH UT_CON ROUTINE LEAVES IN A0.
;
                    MOVE.L    A0,-(A7)        ; SAVE CONSOLE ID ON STACK
                                              ; * NOTE * THIS CHANGES A7
                                              ; BY -4, READY FOR NEXT
                                              ; STACK ENTRY.
;
; CLEAR THE WINDOW (OPTIONAL)
;
                    MOVE.W    #$FFFF,D3           ; INFINITE TIMEOUT
                    MOVEQ     #$20,D0             ; #SD_CLEAR IN D0
                    TRAP      #3
                    BRA.S     NEXTBIT             ; SKIP BLOCK
;
; * NOTE * NORMALLY, WE NEED TO PUT CHANNEL ID IN A0 BEFORE USING THIS
; TRAP TO CLEAR THE WINDOW. HOWEVER, IN THIS CASE IT HAS ALREADY BEEN
; LEFT THERE BY THE PREVIOUS SUBROUTINE
;
; DEFINITION BLOCK.  ADJUST VALUES TO FIT REQUIREMENTS
;
.PBLOCK             DC.B      4                   ; GREEN BORDER
                    DC.B      2                   ; 2 PIXELS WIDE
                    DC.B      0                   ; BLACK PAPER/STRIP
                    DC.B      7                   ; WHITE INK
                    DC.W      100                 ; WIDTH
                    DC.W      100                 ; HEIGHT
                    DC.W      0                   ; X POSITION
                    DC.W      0                   ; Y POSITION
;
; ************************************************************************
```

Listing 4

```
; ************************************************************************
;                              'MESSAGE'
; ************************************************************************
; THIS CODE WILL PRINT A STRING IN A WINDOW.
; THE BASE ADDRESS OF THE STRING MUST BE IN A1.
; THE CHANNEL ID OF THE WINDOW MUST BE IN A0. IF IT HAS BEEN LEFT THERE
; FROM THE PREVIOUS CODE, WE DON'T NEED TO DO ANYTHING ABOUT IT. (IF WE
; DO, WE'LL ONLY PUT A NUMBER WHERE IT ALREADY IS SO IT WON'T MATTER.)
; IF THE CHANNEL ID WAS THE LAST ITEM ON THE STACK IT WILL BE FOUND AT
; (A7), IF THE 2ND FROM LAST AT 4(A7), IF THE 3RD FROM LAST AT 8(A7), ETC
;
.MESS               MOVE.L    (A7),A0     ; OR 4(A7),A0 OR 8(A7),A0, ETC.
                                          ; CHANNEL ID IN A0
                    LEA.L     MESSAGE,A1  ; BASE ADDRESS IN A1
                    MOVE.W    $D0,A2      ; UT_MTEXT VECTOR IN A2
                    JSR       (A2)
                    BRA.S     NEXTBIT     ; SKIP MESSAGE
;
; MESSAGE. ADJUST LABEL, LENGTH AND CHARACTERS TO SUIT
;
.MESSAGE            DC.W      18          ; LENGTH OF MESSAGE
                    DC.B      'THIS IS A MESSAGE:'
```

D1. (In the case of the current job we put -1 into D1.) The actual priority we put into D2.

Console (Listing three)

Jobs almost invariably need at least one console window to input data from the keyboard. The routine 'Console' will open a console window, clear the window (optional), and place the console channel ID on the stack.

The routine uses the UT_CON vector which must be placed in A2 before jumping to its sub-routine. (UT_CON is $C6.) The subroutine will also require the address of a parameter block for the window in A1, and this must be in the following format:

00	byte	border colour
01	byte	border width
02	byte	paper colour
03	byte	ink colour
04	word	window width
06	word	window height
08	word	X origin (top left corner)
0A	word	Y origin (top left corner)

Message (Listing four)

The next thing we would normally want to do is to print some text to the window as a start-up message, or a prompt to input something from the keyboard. The routine 'Message' prints a message to the window.

It requires UT_MTEXT in A2, which it calls as a subroutine. (UT_MEXT is $DO). The subroutine requires the channel ID in A0, and the base address of the message in A1. This particular bit of code could possibly be used several times in a program, and each time it will need modifying. You need to adjust the label, length and characters of the message, and you will have to work out where the channel ID is stored in relation to the current stack pointer. (This only means carefully checking what has been added to the stack since the channel ID was stored.)

Instring (Listing five)

The routine 'Instring' is to input a string from the keyboard and place it in a store called Buffer. Here we run into a Qdos peculiarity which needs overcoming.

If we wish to use this string as a name, say, of a file we wish to open, then Qdos will expect the string to be preceded by a word holding the string length. However, the IO_FLINE routine we use to fetch the string just stores the string in buffer, and leaves the string length (including the LF at the end) in D1. So we precede 'buffer' with a one-word store 'BUF_POS'. We then put D1 in 'BUF_POS' and subtract 1 to remove the LF from the count.

Now, if we need the string preceded by the string count, we consider it as starting from 'BUF_POS'. But if we only need the string without the count, we consider it as starting from 'Buffer'.

Open (Listing six)

The routine 'Open' will open a channel using a name stored in 'BUF_POS'. (We use 'BUF_POS' rather than 'Buffer' be-

Listing 5

```
; ************************************************************************
;                              'INSTRING'
; ************************************************************************
; THIS INPUTS A LINE OF TEXT FROM KEYBOARD AND PLACES IT IN A STORE IN
; RAM CALLED 'BUFFER'.
; THIS STORE HAS TO BE LONG ENOUGH TO HOLD A LINE OF TEXT.
;
.INSTRING           MOVE.L    (A7),A0    ; OR 4(A7),A0 OR 8(A7),A0, ETC.
                                         ; CONSOLE CHANNEL ID IN A0
                    MOVEQ     #BUF_LEN,D2 ; LENGTH OF BUFFER IN D2
                    MOVE.W    #$FFFF,D3  ; INFINITE TIMEOUT
                    LEA.L     BUFFER,A1  ; BASE ADDRESS OF BUFFER IN A1
                    MOVEQ     #2,D0      ; #IO_FLINE IN D0
                    TRAP      #3
;
; IF THIS STRING IS TO BE USED BY QDOS AS, SAY, A CHANNEL NAME, THEN
; WE NEED THE ADDRESS OF CHANNEL NAME IN A0. QDOS EXPECTS THIS NAME TO BE
; PRECEDED BY A WORD HOLDING THE LENGTH. BUT IO_FLINE JUST RETURNS THE
; CHARACTERS, WITH A CHARACTER COUNT (INCLUDING THE LF) LEFT IN D1. WE
; MAKE SURE THE ADDRESS OF BUF_POS IS 2 LESS THAN THE ADDRESS OF 'BUFFER'.
; THEN WE SUBTRACT 1 FROM THE VALUE IN D1 AND PUT IT IN 'BUF_POS'. NOW
; PUTTING THE ADDRESS OF 'BUF_POS' FOR THE ADDRESS OF THE STRING GIVES US
; THE STRING IN THE CORRECT FORMAT.
;
;
                    LEA.L     BUF_POS,A0 ; BUF_POS IN A0
                    SUBQ.L    #1,D1      ; SUBTRACT 1 FROM THE D1 REMOVES
                                         ; THE LF FROM THE STRING COUNT.
                    MOVE.W    D1,(A0)    ; PUT NEW STRING LENGTH IN BUF_POS
;
; ************************************************************************
;                 TO BE POSITIONED AFTER THE CODE
; WE NEED 'BUFFER' FOR THE STRING AND 'BUF_POS' FOR THE STRING COUNT
;
.BUF_LEN            EQU       100        ; LENGTH OF INPUT BUFFER
;
.BUF_POS            DC.W      0
;
.BUFFER             EQU       *
;
;         **** NOTE ****  BUFFER SHOULD COME LAST OF ALL.
;
; ************************************************************************
```

Listing 6

```
; ************************************************************************
;                              'OPEN'
; ************************************************************************
; THIS ROUTINE WILL OPEN A FILE
; WE NEED THE ADDRESS OF CHANNEL NAME IN A0. QDOS EXPECTS THIS NAME TO BE
; PRECEDED BY A WORD HOLDING THE LENGTH. SO WE USE 'BUF_POS' RATHER THAN
; 'BUFFER' AS THE POSITION OF THE START OF THE STRING SO IT STARTS WITH A
; STRING COUNT (SEE 'INSTRING')
;
                    LEA.L     BUF_POS,A0 ; BUF_POS IN A0
                    MOVEQ     #1,D3      ; OPEN OLD SHARED FILE, OR
                                         ; #0 = OLD EXCLUSIVE FILE
                                         ; #2 = NEW EXCLUSIVE FILE
                                         ; #3 = NEW OVERWRITE FILE
                                         ; #4 = OPEN DIRECTORY FILE
                    MOVEQ     #-1,D1     ; JOB ID FOR THIS JOB
                    MOVEQ     #1,D0      ; #IO_OPEN IN D0
                    TRAP      #2
                    TST.L     D0         ; ERROR?
                    BEQ.S     GOT_FILE   ; IF NOT, THEN CONTINUE. ELSE:-
; **** NOTE **** IF THIS ROUTINE IS REPEATED TO OPEN MORE THAN ONE FILE,
; THE NAME 'GOT_FILE' WILL NEED CHANGING FOR EACH OCCURANCE OF THE CODE.
;
; PRINT ERROR MESSAGE
;
                    MOVE.L    (A7),A0    ; OR 4(A7),A0, OR 8(A7),A0, ETC.
                                         ; CONSOLE CHANNEL ID IN A0
                    MOVE.W    $CC,A2     ; UT_ERR IN A2
                    JSR       (A2)       ; PRINT ERROR MESSAGE
;
; GO BACK TO SCREEN PROMPT FOR INPUT
;
                    BRA.S     MESSAGE    ; OR WHATEVER WE CALL THE CODE
;                                        ; FOR THE PROMPT.
;
; IF NO ERROR, THEN THIS CODE IS USED
;
.GOT_FILE           MOVE.L    A0,-(A7)   ; SAVE CHANNEL ID ON STACK
                                         ; * NOTE * STACK POINTER A7
                                         ; CHANGES BY -4
;
; * NOTE * AT THIS POINT THE NEXT CODE WILL PROBABLY BE TO PUT WINDOW ID
; BACK IN A0, TO USE THE WINDOW, OR MAYBE TO CLOSE IT.
;
```

Listing 7

```
; ************************************************************************
;                              'GETHEAP'
; ************************************************************************
; THIS ROUTINE WILL TRY TO ALLOCATE HEAP_ROOM OF COMMON HEAP SPACE AS A
; BUFFER. IF IT CANNOT ALLOCATE THIS MUCH, IT WILL HALVE THE NUMBER AND
; TRY AGAIN. IT WILL KEEP REPEATING THIS UNTIL IT ACTUALLY SUCCEEDS IN
; ALLOCATING AT LEAST 2 BYTES, OR GIVE UP. IF IT SUCCEEDS, IT WILL STORE
```

cause Qdos expects the string to be preceded by a word giving the string length.

We must take care to open the correct type of channel, and this is fixed by the value we put in D3.

0 = old exclusive
1 = old shared
2 = new exclusive
4 = new overwrite
5 = directory

If we are inputting data from an old file, then we would normally make it 'old shared', so other programs can use the file at the same time. But if we are outputting data to a new file, say a serial printer, or a screen window, we would normally make it 'new exclusive' so that it only gets data from this program.

As we might well use this routine more than once in a program, we must also be careful to alter the labels suitably each time it is used.

Getheap (Listing seven)

A useful thing is to use some of the spare, unused ram to speed up program by reducing the number of times we need to access a file on microdrive (or disk). So we allocate a chunk of the common heap (unused ram) as a buffer.

The simplest thing might be to work out (or guess) how much buffer we want, and allocate that amount of common heap for the use of the program. But since we don't know what other programs might be running at the same time, we don't know how much common heap is available. If there is not enough to give us what we ask for, the program will stop with an error message. So the best thing to do is to try to get the amount we want, but if there is not enough, we ask for less, and keep asking for less until Qdos gives us what we ask for.

The routine 'Getheap' keeps halving the amount asked for by using a LSR.L on D1, which contains the amount of ram we are asking for. (If no common heap is available, the program will just stop.)

Stringtr (Listing eight)

The routine 'Stringtr' will transfer strings of bytes from one channel to another. (Both channels will, of course, have to be already opened.) It will continue this until we get complete file transfer, or we get an error. The routine contains alternative bits of code depending on whether we use a small buffer at the end of the program to store the collected bytes, or whether we use some common heap. If we use a small buffer at the end of the program, we can probably use the same one we undoubtedly needed for storing the names of the channels we opened earlier. This means the last part of the routine is redundant, but has been put in for completeness.

This routine makes a lot of use of the stack, with three things stored there; the two channel IDs, and the error return when we input the string (so we can see if we

```
; THE ADDRESS AND NUMBER OF BYTES ALLOCATED IN HEAP_ADDR AND HEAP_LEN.
;
; NOW WE TRY TO GET THAT AMOUNT OF RAM FROM THE COMMON HEAP
;
                    MOVE.L    #HEAP_ROOM,D1 ; BYTES REQUIRED IN D1
.GET_ROOM           MOVEQ     #-1,D2    ; ID FOR THIS JOB IN D2
                    MOVEQ     #$18,D0   ; #MT_ALCHP IN D0
                    TRAP      #1        ; TRY TO GET RAM
                    TST.L     D0        ; ERROR
                    BEQ.S     GOT_AREA  ; YES, THEN CONTINUE. ELSE:-
                    LSR.L     #1,D1     ; DIVIDE BYTES REQUIRED BY 2
                    CMPI.L    #1,D1     ; IF D1<=1 STOP WITH ERROR
                    BGT.S     GET_ROOM  ; ELSE LOOP AND TRY AGAIN
                    BRA.S     JOB_END
;
; HAVING GOT SPACE ALLOCATED, WE MUST SAVE ADDRESS AND NUMBER OF BYTES
; ALLOCATED.
;
.GOT_AREA           LEA.L     HEAP_LENGTH,A1
                    MOVE.L    D1,(A1)   ; SAVE NUMBER OF BYTES
                    LEA.L     HEAP_ADDR,A1
                    MOVE.L    A0,(A1)   ; SAVE AREA ADDRESS
                    BRA.S     NEXTBIT   ; SKIP HEAP ALLOCATION BLOCK
;
; HEAP ALLOCATION BLOCK.
; FIRST WE NEED TO SAY HOW MUCH HEAP_ROOM WE WANT. 4096 BYTES IS 8 MDV
; SECTORS, SO WILL REQUIRE ONE MDV ACCESS FOR EVERY 8 MDV SECTORS IN
; THE FILE.
;
.HEAP_ROOM          EQU       4096      ; HEAP ROOM REQUESTED
.HEAP_ADDR          DC.L      0         ; SPACE FOR HEAP ADDRESS
.HEAP_LENGTH        DC.L      0         ; SPACE FOR LENGTH OF HEAP SPACE
;
; ***************************************************************************
```

Listing 8

```
; ***************************************************************************
;                              'STRINGTR'
; ***************************************************************************
; THIS ROUTINE TAKES A STRING OF BYTES FROM A CHANNEL AND STORES THEM IN
; A BUFFER, UP TO THE LENGTH OF THE BUFFER (BUF_LEN). IT THEN SENDS THE
; STRING TO ANOTHER CHANNEL. IT THEN KEEPS REPEATING UNTIL WE GET EOF.
;
; FIRST WE FETCH THE STRING FROM THE INFILE
.FILE_P             MOVE.L    (A7),A0   ; OR 4(A7),A0, OR 8(A7),A0, ETC.
                                        ; INFILE_ID IN A0
                    MOVEQ     #BUF_LEN,D2 ; IF 'GETHEAP' USED, THEN
                                        ; LEA.L HEAP_LENGTH,A1
                                        ; MOVE.L (A1),D2
                                        ; LENGTH OF BUFFER IN D2
                    LEA.L     BUFFER,A1 ; IF 'GETHEAP' IS USED, THEN
                                        ; LEA.L HEAP_ADDR,A1
                                        ; MOVE.L (A1),A1
                                        ; BUFFER ADDRESS IN A1
                    MOVE.W    #$FFFF,D3 ; INFINITE TIMEOUT
                    MOVEQ     #3,D0     ; #IO_FSTRG IN D0
                    TRAP      #3        FETCH STRING
                    MOVE.L    D0,-(A7)  ; SAVE ERROR RETURN ON STACK
                                        ; * NOTE * STACK POINTER A7
                                        ; CHANGES BY -4
;
; NEXT WE SEND STRING TO OUTFILE
;
                    MOVE.L    4(A7),A0  ; OR 8(A7),A0, ETC.
                                        ; OUTFILE_ID IN A0
                    LEA.L     BUFFER,A1 ; IF 'GETHEAP IS USED, THEN
                                        ; LEA.L HEAP_ADDR,A1
                                        ; MOVE.L (A1),A1
                                        ; BUFFER ADDRESS IN A1
                    MOVE.L    D1,D2     ; IO_FSTRG LEAVES THE NUMBER OF
                                        ; BYTES ACTUALLY FETCHED IN D1
                                        ; SO WE PUT THIS IN D2
                    MOVEQ     #7,D0     ; #IO_SSTRG IN D0
                    TRAP      #3        ; SEND STRING
;
; AT THIS POINT WE NEED TO KNOW IF WE HAVE REACHED THE END OF THE FILE.
; SO WE EXAMINE THE ERROR RETURN SAVED ON THE STACK.
;
                    MOVE.L    (A7)+,D0  ; PUT ERROR RETURN BACK IN D0
                                        ; * NOTE * STACK POINTER A7
                                        ; CHANDES BY +4
                    TST.L     D0        ; ERROR?
                    BEQ.S     FILE_P    ; NO, THEN LOOP FOR NEXT STRING
;
; ***************************************************************************
;                          POSITION AFTER THE CODE
; USE GETHEAP FOR A LARGE BUFFER IN THE COMMON HEAP AREA
; THIS BUFFER WILL HAVE ITS ADDRESS STORED IN HEAP_ADDR AND ITS LENGTH
; STORED IN HEAP_LEN
; OR....
; .BUF_LEN          EQU       100
; .BUFFER           EQU       *
; *** NOTE *** BUFFER SHOULD COME AT THE END
;
; ***************************************************************************
```

have reached the EOF). We must be careful that we know exactly where the stack pointer is at each stage, otherwise we shall be taking the wrong data from the stack!

Close (Listing nine)

Whenever we finish with any channels it is important we close them. If the stack contains other data besides the channel ID we need, then we may have to be a bit careful here. However, if the program is not too complicated (and well ordered) usually by the time we are ready to close channels, all we have left on the stack are IDs of unwanted channels, so there is no problem.

Since closing channels is usually (though not always) followed by killing the job, I have included the line which will jump to the routine to do this.

Endjob (Listing ten)

'Endjob', the last routine in this first article, is the one needed to kill a job, either because it has been completed successfully, or because an error has occurred.

Fatal errors, when detected, should jump to JOB END, and this will give an appropriate error message in channel 0 before killing the job.

A successful completion should jump to END JOB.

In case you feel, after reading this first article, that Assembler Language programming is too difficult, I must point out that later parts are easier. Unfortunately, we can't really start to have a usable program without opening and closing channels to screen, printer, file, etc., and inputting and outputting text. So although these are not the easiest of jobs to program, they really need to come first if we are quickly to get to write a usable program.

Obviously, we will need much more than this if we are going to get very far with our programming. But these are enough for a start, and next time we shall use them to produce a successful program.

```
Listing 9

; ************************************************************************
;                                 'CLOSE'
; ************************************************************************
; THIS ROUTINE WILL CLOSE AN OPEN CHANNEL WHOSE ID IS TOS
;
.CLOSE                  MOVE.L      (A7)+,A0   ; CHANNEL ID IN A0
                        MOVEQ       #2,D0      ; #IO_CLOSE IN D0
                        TRAP        #2
;
; IF THIS IS THE LAST CHANNEL TO CLOSE BEFORE WE KILL THE JOB THEN THE
; NEXT LINE IS NEEDED TO KILL THE JOB. OTHERWISE, LEAVE OUT NEXT LINE.
;
                        BRA.S       END_JOB
;
; * NOTE * IF THE CHANNEL ID WE WANT TO CLOSE IS NOT TOS, WE SHALL HAVE TO
; REMOVE THE OTHER ITEMS ONTO REGISTERS (OR RAM) TO GET TO THE ITEM WE
; WANT, USE IO_CLOSE, AND THEN PUT THEM BACK ON THE STACK.
;
; ************************************************************************
Listing 10

; ************************************************************************
;                                 'ENDJOB'
; ************************************************************************
; THE FIRST PART OF THIS ROUTINE SHOULD ONLY BE USED WHERE AN ERROR IS
; DETECTED BEFORE A SUITABLE WINDOW HAS BEEN OPENED. IT WILL GIVE AN
; ERROR MESSAGE IN CHANNEL 0, OR IN CHANNEL 1 IF CHANNEL 0 IS NOT
; AVAILABLE.
;
.JOB_END                MOVE.W      $CA,A2 ; UT_ERRO in A2
                        JSR         (A2)
;
; THE NEXT PART IS THE NORMAL ENDING OF A JOB BY KILLING IT.
;
.END_JOB                MOVEQ       #5,D0      ; #MT_FRJOB IN D0
                        MOVEQ       #-1,D1     ; ID OF THIS JOB IN D1
                        TRAP        #1
;
; ************************************************************************
```


QUANTA

Independent QL/Thor Users Group

Worldwide Membership is by subscription only, and offers the following benefits:
Monthly Newsletter – up to 40 pages
Massive Software Library – mostly FREE!
Free Helpline and Workshops
Regional Sub-Groups. One near you?
Advice on Software & Hardware problems
Discounts from most major suppliers
Subscription just £14 for UK members
Overseas subscription £17

Barclaycard: Visa: Access: Mastercard

* Now in our EIGHTH successful year *

Further details from the Membership Secretary

Bill Newell
213 Manor Road
Benfleet
Essex. SS7 4JD
Tel (0268) 754 407

QUBBESoft P/D

Public Domain and Shareware Software
For The
Sinclair Q.L.
S.A.E. (A5) for Software Catalogue

Copying Fee 50p

Copying Fee 30p

C68 Public Domain 'C' Compiler

Disk 1: (Main System Disk) Contains the various passes of the Compiler, Header files and the Libraries.

Disk 2: (Utils & Doc's) Contains Utility Programs and Documentation for the C68 system. Also includes a 'C' Tutorial.

Disks 3-7: (Source Code 1-5) Contains Source Code for following:- C68 Compiler, CC, CPP, AS68, LD, Generic Part of Standard 'C' Library, QDOS Specific Part of 'C' Library, Additional Libraries and Utilities such as MAKE and SLB.

Copying fee: £2.50p (User Supplies Disks) or £5.00p (Pre-Copied Disks)

MicroEMACS V3.9p Text Editor

Disk 1: (System & Doc's) Contains
EMACS, The main program.
EMACS_RC, Resource file.
EMACS_HLP, Help file.
EMACS_MSS, User guide.
EMACS_TUT, Tutorial file.
EMACS1_doc to EMACS13_doc
EMACSA_doc to EMACSF_doc is User Guide in _doc form

Disk 2: (Source Code) Contains All the relevant Source Code for MicroEMACS V3.9p.
Also includes documentation on the latest update, and how it has been re-compiled using the C68 Compiler.

Copying Fee: £1.00p (User Supplies Disks) or £2.00p (Pre-Copied Disks)

38, Brunwin Road, Rayne, Braintree, Essex. CM7 5BU. Tel No. (0376) 47852
Cheques/P.O's Payable to "K.Dunnett" C68 P&P 50p, EMACS P&P 30p

DIY TOOLKIT

Simon Goodwin shows how you can turn QL task files into the keywords of your dreams.

Task Commander turns Qdos task files into resident procedures. Tasks run instantly when you type your chosen commands. This is not just a great way to extend SuperBasic; it also shows how to write code that creates, loads and activates QL tasks.

The program uses extended SuperBasic to add a header, giving a task a name, like a built-in or Toolkit command. The resultant code remains a fully-fledged task, so you can launch and run it at the same time as Basic, perhaps passing channels or other parameter data for it to work on.

All *Task Commander* files can run from rom, regardless of the compiler used to generate the task. The task code is copied into the transient program area before running. Tasks can be up to one megabyte in size; even the largest should start in under a second.

As usual, the programs are tested on Sinclair JM roms, *Minerva* Qdos 1.64 and Argos 6.41. As far as I know, files should be totally compatible between rom versions.

I have often been asked for some way to turn compiled tasks into keywords. At last I've done it, and my systems are now rapidly filling with useful new keywords as a consequence. I took my cue from Quanta boss Phil Borman, who kindly sent me a disk of his program MAKE_KEYWORD_BAS earlier this year. It uses Turbo Toolkit to add a short header to *Turbo* tasks.

Resulting files can be loaded into memory like any Toolkit command. The chosen keyword starts the task; for instance, you an type BLAST instead of EXEC_W FLP1_BLAST.

Phil's routine works well, but has limitations. The calling task waits until the new task is finished; there is no way to pass parameters, and keyword names cannot exceed seven characters or Name Table corruption might result.

The DIY Task Commander needs a little more ram than Phil's MAKE_KEYWORD. It is safe with any length of name, launches tasks even more quickly, and allows full Qdos task parameters. It can start a task from SuperBasic without stopping the interpreter. Just add a comma at the end of the command, to get the effect of EXC rather than EXEC_W.

Parameter passing between tasks was not mentioned in the original 1984 Qdos manual, but it is documented in section 24 of QJumps *Supertoolkit 2 Guide*. The standard method passes any number of channels on the task's stack, indicated by a count word followed by the corresponding number of long word channel identifiers. It can also pass a string of up to 32766 bytes.

This parameter-passing scheme allows pipes and filters, in the style of the Unix operating system. It is implemented by many QL compilers, including *GST C, Pro Pascal, Qliberator* and *Turbo.*

For instance, QLib assigns channels to #0, #1, #3, #4 and so on; Turbo tasks assign channels in order downwards from #15. Compiled programs read the string parameter by refering to a pre-assigned name like ARGV, CMD$ or OPTION_CMD$. Your compiler manuals should have details.

It appears that this scheme was intended from the start, as even the AH rom CJOB routine puts two zero words on the stack by default; these signify no channels and no characters.

The first half of **Listing one** evaluates, checks and passes the parameters on to the task. It is important to validate them all before allocating memory for the task; we don't want to waste time or memory if an error is inevitable.

The parameters are simple: any number of open SuperBasic channel numbers, followed by an optional string. The last parameter is taken as a string unless it has a # prefix; thus you can coerce a string to a channel number, and pass nothing but channels, or pass any number, coerced to a string.

In five years DIY Toolkit has demonstrated many styles of QL parameter scanning, allowing options, defaults, and the full range of data-types; now I'm going to show you how to do it backwards! Listing one scans from right to left. This is the easy way to pick up the comma and string parameter which may be at the end of the statement. Everything else must be channels.

This sequence means that the SuperBasic value-fetching routines put the parameters in the ideal order on the RI stack, so they can be copied *en bloc* to the data area of the task before it is activated. There may be anything from zero to a dozen or more parameters; typically you might pass a short string and zero, one or two channels, like this:

```
CHURN "flp1_help/flp2_report",
SPLICE #0,#3,"temp_merge"
```

If no explicit channels are specified, the DIY code passes the identifier of the Default SuperBasic channel #1. The task does not have to use this, but may find it convenient. The DIY Toolkit USE keyword, from February 1988 and Volume C, can make this the default channel in the compiled task, too; just add USE 0 for Qlib or USE 15 for Turbo.

The identifier of channel #1 upsets Psion *Xchange,* which does not expect to be passed any channels, and gives a bad parameter error otherwise. To fix this, change the first and last numbers in line 870 of **Listing two;** replace 26696 with 28232, and change 26122 into 24586 at the end of the line. This swaps BNE.S for BRA.S, so that #1 is not passed unless it is an explicit parameter.

Remember that Qdos cannot allow one task to use a channel while another is waiting for data from the same channel. It is unwise to pass SuperBasic windows #0,#1 or #2 to tasks if you intend to use the trailing comma to allow SuperBasic to carry on; output may be delayed or muddled. Programmers can check if a channel is busy by calling CHBASE, from DIY Toolkit, Volume Q.

The machine code in Listing one demonstrates some tricky parameter handling, and shows how tasks are created and activated. Listing one is the assembler source of the code added to each task, tested with *Devpac.*

You can type the code of Listing one into any QL assembler, and customise it for

your needs. Minor syntax variations mean you may need to suffix 'A' to a few address instructions, and omit '\w' when reading word vectors.

The Basic loader configures the code to suit each task, inserting the name and address of the Resident Procedure at PARAMS, patching the code and data sizes at MAKE_TASK, and tacking the code of the task onto the end, at TASK_CODE.

Task Commander is very useful for small tasks, unless you have plenty of memory. Some program tasks modify their code after loading, so you need two copies in memory to run one. The master copy needs no data space, but about 280 bytes are required to process the name and parameters, and launch the task.

QL tasks are small compared with memory capacity; most packages work in under 90 K, to suit the original QL, yet the long-established standard expansion of 640 K ram has been bettered by the 896 K *Trump Card*. Now the standard is set by the two megabyte *Gold Cards*, multi-megabyte Thors and Qdos emulators for ST and Amiga

Memory is cheap now, be it rom or ram. Task Commander is best if you have ram to spare, or itch to fill one of Quanta's 192 K Eprom expansion boards, or the 32 K rom sockets on CST's RAM+.

If all the tasks you use are re-entrant you could conserve memory by passing the address of TASK_CODE to MT.CJOB in A1, setting D2 to zero. This won't suit all compilers, but it saves making a copy of the code each time you want to run a task.

The code labelled ADJUST_A3 prevents the parameters being copied if there is not enough room for them in the data space.

Notice that the SUB.W D0,A3 instructions implicitly extend D0 to a long-word value before subtracting it from A3. This is important, as A3 invariably exceeds 32767. Two steps are used as D0 is a count in words, and A3 addresses bytes.

Assembler programmers soon learn the importance of keeping values in processor registers; this policy avoids repeated access to memory, and is vital for fast QL code. When there are many variables in use it may be difficult to assign registers.

As usual, the innermost loop is the one to optimise first. Values therein are used most, so they must be the most accessible. But which registers can be used? This is my approach.

Aim to preserve useful register values for your code and the routines it calls. The rom uses A0 for the Channel ID, D1 for the Task ID, D3 for timeout; other defaults were listed in July's DIY Toolkit Volume T. There are 16 general-purpose registers, rather than two or three of early processors; it is still easy to run out, especially when writing SuperBasic extensions.

A7 and A6 are pre-defined, delimiting the subroutine stack and the start of Basic memory; A1, A3 and A5 point at parameters, while A0 and A2 get clobbered by the CA.GT parameter routines, along with D0, D1, D2, D3, D4 and D6. The only spare registers are A4, the interpreters's pseudo-program counter, D5, and D7, which must sometimes be zeroed to avoid a maths bug in early Sinclair roms.Task Commander uses D5 and D7 to pass parameter details to the loader. D5.L and D5.W are used for two distinct purposes. The sign of D5.L distinguishes EXEC from EXEC_W, while D5.W holds the length of the parameter string.

When it gets tricky to allocate optimal register numbers, I find changes are kept to a minimum if I program using names like D8 and D9 as well as the 'real' registers D0 to D7. These must be replaced before the program is assembled, but they help me see just what is needed, and where.

Qdos saves and restores 68008 register values in the job header, immediately before the code space, as it swaps from task to task. The instruction before ACTIVATOR sets JB.A7, the stored value of the

```
DIY TOOLKIT - OCTOBER 1991 - Listing 1, page 1 of 2

* QL WORLD DIY TOOLKIT - TASK COMMANDER ROUTINES
* Version 0.9, Copyright July 1991 Simon N Goodwin
*
* BP.INIT expects a PROC name & address to be inserted at PARAMS
*
setup       lea.l    params,a1        Point at the details
            move.w   $110\w,a2        Fetch BP.INIT vector
            jmp      (a2).            Add a resident procedure
params
*
not_open    moveq    #-6,d0           CHANNEL NOT OPEN error code
bad_return  rts
*
* TASK_CMD_NAME { # CHANNEL% , }   [ PARAMETER$ ]   [ , ]
*
taskcmd     moveq    #0,d7            Default channel count
            moveq    #0,d5            Default parameter string: ""
            cmpa.l   a3,a5            Check parameter pointers
            beq.s    make_task
            btst     #4,-7(a5,a6.l)   Is there a trailing comma?
            beq.s    checked
            bset     #31,d5           Aha, it's EXEC not EXEC_W
            tst.b    -8(a5,a6.l)      Is there a data type too?
            bne.s    checked
            subq.l   #8,a5            Forget the last 'parameter' ,
            cmpa.l   a3,a5            Is there anything else?
            beq.s    make_task
checked     tst.b    -7(a5,a6.l)      Does the last one have a # ?
            bmi.s    get_chans        Positive if not, assume $
*
* Pick the string parameter off the end of the parameter list
*
get_string  movea.l  a3,a4            Note location of parameters
            lea.l    -8(a5),a3        Focus on the last one
            movea.w  $116\w,a2        Pick up CA.GTSTR vector
            jsr      (a2)             Get string onto RI stack
            bne.s    bad_return
            move.w   0(a1,a6.l),d5    D5 is text length, 0-32767
            bne.s    find_chans
            addq.l   #2,a1            Ignore nulls on the RI stack
find_chans  cmpa.l   a3,a4            Is there anything else?
            beq.s    make_task
            movea.l  a3,a5            Find the other parameters
            movea.l  a4,a3            Recall pointer to first
get_chans   movea.w  $118\w,a2        Fetch CA.GTLIN vector
            jsr      (a2)             Stack some long integers
            bne.s    bad_return
            move.w   d3,d7            Update channel count
*
* Check each channel number before allocating task memory
*
            movea.l  a1,a4            Keep RI stack base offset
check_chan  move.w   2(a1,a6.l),d0    Get LOW word of parameter 1
            bmi.s    not_open
            mulu     #40,d0           Each table entry is 40 bytes
            add.l    48(a6),d0        Add channel base offset
            cmp.l    52(a6),d0        Check it is within the table
            bge.s    not_open
            move.l   0(a6,d0.l),d0    Pick up the Qdos channel ID
            bmi.s    not_open
            move.l   d0,0(a1,a6.l)    Store ID in place of #number
            addq.l   #4,a1            Advance to next parameter
            subq.w   #1,d3            Process all in turn
```

stack pointer in the job header. The default value set by MT.CJOB points at two zero words at the end of the data space; we need to move A7 down to address the parameters.

The main Task Commander program is written in SuperBasic. A version based on SuperToolkit extensions appears in Listing two. It merges the task with the DIY set-up code, adding the chosen command name. It uses common Toolkit commands, and can easily be converted; I have versions that use DIY Toolkit, Turbo Toolkit and SuperToolkit commands.

All three versions – plus documentation, examples and assembler source – have been added to the File Tools disk, Volume F of DIY Toolkit. Send £7 for the complete volume, which includes the *Customkit* toolkit file combiner, and a utility to create eprom headers. make payment out to **DIY Toolkit** (not to CGH Services), **Cwm Gwen Hall, Pencader, Dyfed, Cymru SA39 9HA,** or phone Richard on (0559) 384574 1 pm - 9 pm.

There are now 17 volumes. Each costs £3 plus £4 per order to cover disks, post and processing. Please send one formatted cartridge per volume if you need the files on microdrive.

Listing Two uses a handful of Supertoolkit commands. Turbo Toolkit fans may prefer to replace FTEST and FOP_IN with DEVICE_STATUS; ALCHP and RECHP correspond to ALLOCATION and DEALLOCATE, while FDAT and FLEN mimic DATASPACE and POSITION at the end of the file.

The DIY version uses GetHEAD, RESERVE and DISCARD extensions, supplied with Volume F along with the source for both other versions. The differences stem from variations in the commands and error-trapping facilities of each toolkit.

The Turbo-Toolkit variant has the most comprehensive checks, but may stop if you run out of memory or space on your drive, or try to write to a 'read only' medium.

Messages appear if the source file, containing the task, is absent, too long or an odd length. Odd length files crash EXEC. The code copies 16 bytes at a time, in up to 65536 steps, moving up to a megabyte at high speed.

The task data space must be at least 16 bytes; most tasks use far more, so that the message 'data space too small' probably means your source is not a task file at all. Parameters need at least eight bytes, and the fast copier may overrun by up to 16 bytes when copying the task code; it's faster that way.

The destination file contains the new command code, which you can link in the usual way with LRESPR, LINKUP or RESPR, LBYTES and CALL. It needs slightly more space than the source file.

The code loader is not the usual DIY Toolkit routine; it does not make a temporary file, and stores decimal words rather than hex strings. DATASTORE is shorter, faster and more reliable than the hex loader which Marcus Jeffrey bequeathed to this column in 1987.

Each DATA line starts with a checksum, followed by a count of the number of 16 - bit values thereafter. The checks for each line detect extra or missing items and report the line number if a problem is found, so you need only correct that line. The exception is 'DATA length error', which indicates that the total of the 'count' values does not add up to the expected number of words. If you get this report, check the second value in each DATA line.

All the data is numeric, as many people have number key-pads and I wish to avoid confusion between the letter B and the digit 8. The digit zero appears with a slash throughout, to distinguish it from letter O.

Users with their own hex editor might prefer the hex representation. It is easy for me to use either format, as I have software tools to create and check the DATA, so let me know what you prefer, care of *QL World.*

You may need to comment out the RESTORE at line 600 before compiling Listing two. It is only needed when you first

```
DIY TOOLKIT - OCTOBER 1991 - listing 1, page 2 of 2
* D7.W is No. of channels; D5.W is text length; D5.L<Ø for EXEC
*
make_task   move.l   #Ø,d3            Data space - to be patched
            move.l   #2,d2            Code space - to be patched
            moveq    #-1,d1           Owner is this task
            suba.l   a1,a1            A1 is Ø; put task in TRNSP
            moveq    #1,dØ            MT.CJOB trap key
            trap     #1               Create a task
            tst.l    dØ               Did that work?
            bne.s    return_err
            lea.l    8(aØ,d2.l),a4    A4 -> start of data space +8
            lea.l    -8(a4,d3.l),a3   A3 points past end of dspace
*
* Copy the code into the task area; ignore slight over-runs
*
            lsr.l    #4,d2            Divide by 16 (roughly)
            move.l   aØ,a1            Copy pointer to code space
            lea.l    task_code,a2     Find the code
copy_code   move.l   (a2)+,(a1)+      Repetition improves speed
            move.l   (a2)+,(a1)+
            move.l   (a2)+,(a1)+      Move 16 bytes per DBRA step
            move.l   (a2)+,(a1)+
            dbra     d2,copy_code     Copy up to 1 Megabyte, fast
*
* Copy the parameters from RI to JB.SP, counting in 16 bit words
*
            move.w   d7,dØ
            add.w    dØ,dØ            Number of channel ID words
            tst.w    d5               Check string length
            bne.s    string
no_string   clr.w    -(a3)            Stack a null length
            bra.s    adjust_a3
string      addq.w   #3,d5            Include length + an odd byte
            lsr.w    #1,d5            D5 := string size in words
            add.w    d5,dØ            DØ := stack words required
adjust_a3   sub.w    dØ,a3            Move A3 down to make room
            sub.w    dØ,a3            DØ is in words, A3 is bytes
            cmpa.l   a4,a3            Ensure room for parameters
            bcs.s    activator        Pass nowt if they won't fit
            movea.l  a3,a4            Save bottom address
            move.l   $58(a6),a1       Recall the BV.RIP offset
            bra.s    copy_loop        Copy nothing if DØ=Ø
copy_pars   move.w   Ø(a1,a6.l),(a3)+
            addq.l   #2,a1            Advance up the RI stack
copy_loop   dbra     dØ,copy_pars
*
* If CHANS=Ø, stack 1.W and the ID from #1, else stack CHANS.W
*
chan_check  tst.w    d7               Check channel count
            bne.s    stack_ok
            moveq    #1,d7            Pass default, ID of #1
            movea.l  48(a6),a2        A2 is offset to BV.CHBAS
            move.l   40(a6,a2.l),-(a4)  Presume that #1 exists
stack_ok    move.w   d7,-(a4)         Stack the channel count
            move.l   a4,-12(aØ)       Set JB.A7 in task header
*
* Activate the task
*
activator   moveq    #16,d2           Default priority (Ø-255)
            moveq    #-1,d3           Timeouts Ø = EXEC, -1 = WAIT
            tst.l    d5               Did command end with a comma?
            bpl.s    time_set         If not, timeout is -1, wait
            moveq    #Ø,d3            EXEC; no need to wait
time_set    moveq    #10,dØ           MT.ACTIV trap key
            trap     #1
return_err  rts                       Return error code (if any)
*
task_code   end                       Task code follows here
```

run the program to check the data, and Turbo or Supercharge object to the variable parameter. Once the data has been checked all you need is a single RESTORE at the start of the program.

After just a few weeks I find the Task Commander is indispensable. The compiled task, TASCOM, is itself a resident command on my QL and Thor; every new task is a candidate for installation. I am grateful to Phil Borman for sending me down this track.

The scope for new DIY Toolkit extensions seems to be narrowing, and I welcome new directions. Ideal topics involve short, original machine - code routines which can be used as 'building blocks' in other programs. Please send your suggestions to me.

```
DIY TOOLKIT - OCTOBER 1991 - listing 2, page 1 of 2

100 REMark Program to convert any Task into a Resident Procedure
110 REMark SuperTOOLKIT 2 version 1.1, with full parameter passing
120 maxcode=2^20 :REMark DIY fast load limits tasks to 1 Megabyte
130 :
140 WINDOW 400,150,36,30 : BORDER 2,0,7 : PAPER 0 : CLS
150 PAPER 2 : INK 7,4 : CSIZE 2,1 : CLS 3
160 PRINT "TASK COMMANDER QL file converter"
170 PAPER 7 : INK 2 : CSIZE 1,0 : CLS 3
180 PRINT "    Written by Phil Borman & Simon N Goodwin."\\
190 INK 4 : PAPER 0
200 :
210 INPUT ' Source file >'!source$
220 INPUT ' Destination >'!destin$
230 INPUT ' Keyword     >'!name$\\
240 IF name$="" : PRINT " Huh?" : STOP
250 flag=FOP_IN(source$)
260 IF flag<0 : PRINT ' Error reading'!source$ : REPORT #1,flag : STOP
270 file_length=FLEN(#flag)
280 file_data = FDAT(#flag)
290 CLOSE #flag
300 IF file_length>maxcode OR 2*INT(file_length/2)<>file_length
310   PRINT source$!"has odd or excessive size" : STOP
320 END IF
330 IF file_data<16 : PRINT !source$!"data space too small" : STOP
340 buffer=ALCHP(file_length+280+LEN(name$)) : NL%=LEN(name$)
350 LET b=buffer : DATASTORE 740 :REMark Store BP.INIT code
360 offset=6+NL%-(NL% && 1) :REMark Find the PROC code
370 POKE_W b,(NL%+8) DIV 8 : POKE_W b+2,offset+8 : POKE b+4,NL%
380 FOR i=5 TO NL%+4 : POKE i+b,CODE(name$(i-4)) && -33
390 b=b+offset : POKE_L b,0 : POKE_W b+4,0 : b=b+6 : base=b
400 FOR lnum=760 TO 890 STEP 10 : DATASTORE lnum
410 IF base+254<>b : RECHP buffer : PRINT " DATA length error!" : STOP
420 POKE_L base+124,file_data : POKE_L base+130,file_length
430 LBYTES source$,b :REMark Load task to end of code
440 flag=FTEST(destin$) : size=file_length+b-buffer
450 IF flag=0
460   PRINT destin$!'already exists - Delete (Y/N) ? ';
470   k$=INKEY$(#1,-1) : PRINT k$
480   IF k$=='y' THEN DELETE destin$ : flag=-7 : ELSE flag=-8
490 END IF
500 IF flag=-7
510   SBYTES destin$,buffer,size
520   PRINT \\name$!"keyword code has been created in"!destin$
530 ELSE
540   PRINT ' Error writing'!destin$ : REPORT #1,flag
550 END IF
560 RECHP buffer : STOP
570 :
580 DEFine PROCedure DATASTORE(line%)
590 LOCal i, x, words%, key%, checksum%
600 RESTORE line% :REMark Interpreters only (initial checking)
610 key%=3 : READ checksum%, words%
620 FOR i=0 TO words% * 2 -2 STEP 2
630   READ x : POKE_W i+b,x
640   IF x>32767 THEN x=x-65536
650   key%=key% ^^ x ^^ i
660 END FOR i
670 IF key%<>checksum%
680   PRINT "Error in DATA line"!line% : RECHP buffer : STOP
690 END IF
700 b=b + words% * 2
710 END DEFine DATASTORE
720 :
730 REMark BASIC Initialisation code
740 DATA 14403,5,17402,8,13432,272,20178
750 REMark Resident procedure code
760 DATA 1777,9,28922,20085,32256,31232,48075,26478,2101,4,59641
770 DATA 9730,9,26384,2245,31,18997,59640,26118,20877,48075,26454
780 DATA 8803,9,18997,59641,27422,10315,18413,65528,13432,278,20114
790 DATA 21657,9,26314,14897,59392,26114,21641,47563,26422,10827,9804
800 DATA -18292,9,13432,280,20114,26290,15875,10313,12337,59394,27558
810 DATA -28691,10,49404,40,53422,48,45230,52,27800,8246,2048,27538
820 DATA -31025,11,9088,59392,22665,21315,26332,8780,9788,0,0,9276,0
830 DATA -28043,9,2,29439,37833,28673,20033,19072,26218,18928,10248
840 DATA -3989,10,18420,14584,59530,8776,17914,94,8922,8922,8922,8922
850 DATA 16476,8,20938,65526,12295,53312,19013,26116,16995,24582
860 DATA -17263,9,22085,57933,53317,38592,38592,47052,25894,10315,8814
870 DATA 26696,9,88,24582,14065,59392,21641,20936,65528,19015,26122
880 DATA 18685,9,32257,9326,48,10550,43048,14599,8524,65524,29712
890 DATA 20555,7,30463,19077,27138,30208,28682,20033,20085

((END))
```


NEWSAGENT ORDER FORM

By placing a regular order with your newsagent you can be sure of receiving **Sinclair QL World** every month. Complete this coupon and pass it on to your newsagent.

Dear NEWSAGENT,

Please reserve/home deliver _____ copies of **Sinclair QL World** every month for:

Mr/Mrs/Miss .....

Address .....

.....

Telephone No .....

Sinclair QL World is published on the 3rd Thursday of every month. Any distribution queries to: **IPC Marketforce, Kings Reach Tower, Stamford Street, London. SE1 9LS. Tel: 071 261 5000.**

NOW WITH DISKS

MICRODRIVE EXCHANGE

B = SuperBasic; A+O = assembler and object code; M+B = machine code and Basic loader; A+B+O = assembler and Basic loader and object code; S = supercharged; L = QLiberated; f1 = monitor mode; f2 = TV mode.

Established QL suppliers Sector Software are duplicating and distributing Microdrive Exchange programs, and supplying 3.5 inch disks. Please read the small print carefully, as different action is needed depending on whether you want mdv or disk copies.

I understand that Sector Software undertake only to supply these programs, and accept no liability for operation as defined by the authors. QL World cannot supply information about the programs other than that originally printed or supplied.

Any article reprints must be ordered and paid for separately at £2 each (£3 overseas) from *Sinclair QL World* at the Editorial address.

HOW TO ORDER

The Microdrive Exchange will copy the programs of your choice onto Sinclair microdrive cassette or 3.5 inch disk. Please note we can only supply ONE PROGRAM PER MDV OR DISK.

Orders for programs on DISK can only be supplied on the MDX's own disks, priced at £1, one disk per program ordered. Please do NOT send disks. Please state any specific density requirement for your set-up. Microcassettes cost £2.60 each.

Please send your orders to Sector Software, 39, Wray Crescent, Ulnes Walton, Leyland, Lancs. PR5 3NA. Tel. 0772 454328. Access/Visa accepted.

ORDER FORM

Value of programs ordered £.....

I require one microcassette/one disk for each program ordered (delete as applicable)

No. of 3.5in. disks required @ £1 each £.....

No. of mdvs required @ £2.60 each £.....

Post and packing £1.00

17½% VAT £.....

TOTAL £.....

PROGRAM IDENTITY NUMBERS:
/ / / / / / / / / / /

NAME .....

ADDRESS (BLOCK CAPITALS PLEASE) .....

.....

.....

.....**Postcode**.....

ACCESS/VISA NUMBER IF APPLICABLE:
☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐

Please copy on to mdv or disk the programs which I have indicated. I enclose a cheque/PO/Access or Visa number to the total of £..... payable to Sector Software.

THE PROGRAMS

1. LEAGUE SECRETARY by C.B. Storey (B) £3

You enter the match results and this program updates the league tables. Suitable for any sporting league organised on the lines of the Barclays Football League. Mdv only at present.

2. THE DOUBLE by P.G. Ives (Bf2) £4

A large football strategy game. You manage a team through four divisions, buying and selling, boosting morale through the league and F.A. Cup season.

3. SUPERBREAKOUT by R. Davison (M+B) £2

Fast m/c version of the classic bat, ball and wall game. Optional double bats and/or balls.

4. SPACE PODS by Simon Quinn (M+B) £3

Your lone ship must protect six energy pods against the aliens. Machine code. *QL World,* December 1987.

5. GRAPHIC WRITER by S.M. Walker (B) £2

A graphic design program which can save your pictures as SuperBasic commands for use in other programs. *QL World,* December 1987.

6. ZAPMAN by L. Miles (M+B) £3

Fast-action m/c version of the Pacman genre. Variable skill levels and maze formats.

7. SPACE INVADERS by Paul McKinnon (M) £3

Very fast, challenging version of the classic, with ugly aliens and protective shields.

8. SPELLED by Timo Salmi (T) £3

A complete spelling checker for Quill_list files, 7,500 words automatically expandable. Requires two cartridges and 512K expansion.

9. ADVENTURE PLAYTIME by A. Pemberton (B) MDV only £3

An extensive adventure where you must complete tasks for the inhabitants of a strange land. Coded messages and hints included.

10. YAHTZEE/GIRO by Jason Price/Henry Wrighton (B) £3

The popular dice game with instructions and graphics, for one or more players. *QL World* November 1987. Also space-disaster rescue game, in zero-gravity. *QL World* December 1989.

11. LOCK AND KEY by Henry Wrightson (B) £3

Unlock the secrets of eight screens of platforms, ladders, keys, poison weeds and mixed fruit — against the clock. *QL World* February, 1990.

12. RADAR by Nigel Ford (B) £3

You are control, monitoring the skies, checking aircraft, scrambling jets to intercept UFO's and shooting down enemy aircraft.

13. TAKTIX by Nigel Ford (B) £3

Six or more can play the computer in a fierce game of European conquest. Put aside at least an hour. *QL World* July, 1988.

14. BRIDGE by Peter Etheridge (B) £4

Excellent version, including accurate bidding, automatic or manual card play, replay hands, save and loads more.

15. CONQUEST by Andrew Pritchard (T) £4

"Superb graphics, with lots of original ideas. The best strategy game I've reviewed for *QL World.*"

16. SPEEDMIND by William Henderson (B) £3

A mastermind-style game played with coloured pages. You have 12 attempts at breaking the code against the clock. *QL World,* January 1988.

17. STELLARIS by David Carmona (Bf1) £4

Real-time space adventure against the computer, including economic simulations, lunar landing and superb graphics. *QL World,* June 1987.

18. BUSINESS GAME by David Smith (B) £4

A business simulator for any number of players, human or computer. The winner is the one who makes the most money! Networking advice from author. See *QL World,* April 1989.

DBQL

In the fourth part of his relational database, Tom Ashcroft implements multiple file opening.

```
1000 start
1010 REMark ************************ START
1020 DEFine PROCedure start
1030 MODE 4
1040 OPEN #3,ser1
1050 OPEN #4,con_480x30a10x0
1060 WINDOW 480,206,10,30
1070 WINDOW#0,480,20,10,236
1080 nemo=1:chan=1:alias=1:databases=0:lprint=0
1090 ann=1:afields=1:achars=1:adeleted=0:aindices=0:ac=1:aresave=0:
aoldfile=0:apac=0:af$="":aactive=1:aselected=0
1100 bnn=1:bfields=1:bchars=1:bdeleted=0:bindices=0:bc=1:bresave=0:
boldfile=0:bpac=0:bf$="":bactive=1:bselected=0
1110 cnn=1:cfields=1:cchars=1:cdeleted=0:cindices=0:cc=1:cresave=0:
coldfile=0:cpac=0:cf$="":cactive=1:cselected=0
1120 PAPER 0:PAPER #0,0:PAPER#4,0:INK 7:INK #0,7:INK#4,4:CLS:CLS#0
1130 menu
1140 END DEFine start
1150 REMark ************************ MENU
1160 DEFine PROCedure menu
1170 CLS#4
1180 PRINT#4,"CReate    OPen    ENter   Goto   Back    Next    FInd
MOre   LOokup   AMend   PRinter    DELete    SElect   DEselect   LIst
USE   INdex    PAck    QUIT"
1190 IF lprint THEN PAPER#4,4:INK#4,7:PRINT#4," Is the printer ready?
":PAPER#4,0:INK#4,4
1200 END DEFine menu
1210 REMark ************************ ENTER
1220 DEFine PROCedure en
1230 REPeat entrygroup
1240 SELect ON alias
1250    =1:enter a$,ann,afields,aresave,achars,aindices,andx$, aactive
1260    =2:enter b$,bnn,bfields,bresave,bchars,bindices,bndx$, bactive
1270    =3:enter c$,cnn,cfields,cresave,cchars,cindices,cndx$, cactive
1280 END SELect
1290 ssave:lload
1300 IF q$="©" THEN EXIT entrygroup
1310 END REPeat entrygroup
1320 CLS:menu
1330 END DEFine en
1340 DEFine PROCedure
enter(x$,nn,fields,resave,chars,indices,ndx$,active)
1350 CLS:CLS#4:resave=1
1360 PRINT#4," Type information field by field      Key ENTER at the
end of each field"\"                                Shift/ESC to quit"
1370 REPeat entryloop
1380  nn=nn+1:CLS:PRINT"Record no ";nn
1390  FOR j=1 TO fields STEP 19
1400   FOR k=j TO j+18
1410    IF k>fields THEN EXIT k
1420    PRINT x$(0,k,4 TO LEN(x$(0,k)));TO 10;";"
1430   END FOR k
1440   FOR k=j TO j+18
1450    IF k>fields THEN EXIT k
1460    AT (k-j)+1,12;
1470    IF x$(0,k,3)="N" THEN
1480      q$=intake$:IF q$="" THEN q$="0"
1490    ELSE
1500    IF x$(0,k,3)="D" THEN
1510      q$=checkdate$:IF q$="" THEN q$="000000"
1520      q$=date_rev$(q$)
1530    ELSE
1540      INPUT q$
1550    END IF :END IF
1560      IF q$="©" THEN nn=nn-1:EXIT entryloop
1570      x$(nn,k)=q$
1580   END FOR k
1585   AT 1,0:CLS 2:CLS 3
1590 END FOR j
1600 x$(nn,0)=" "
```

The ability to open more than one file simultaneously is a great step forward in the versatility of a database system. It makes it possible to take information from one file to enter in another, or to combine with data from another file or files to form a composite report. Again, two or more files can be linked by a common field so that each contains information about one set of relationships and can be related to the other. For example, a car hire company might have one database containing information about customers such as name, address, date and duration of hire, method of payment, previous hirings etc., and a second database containing information about the fleet such as make, model, engine capacity, date of purchase, date of next service etc. These two databases might have a field, such as a 'car registration number', which is common to both. One file contains everything relating to the customer and the other everything relating to the car, linked by the common field, hence the name **relational database.** To use such a system it must be possible to manipulate and move between databases with ease.

Two ways

Turning to DBQL, there are two possible ways of implementing multiple files. The easiest is to create A$ as a multidimensional array. DIM a$ (5,10,15,20) would provide for five arrays, each with 10 records consisting of 15 fields, each 20 characters in length. The drawback to this system is that all the arrays are automatically dimensioned to the same size and cannot be varied, and the size would obviously have to accommodate the largest number of records, fields and characters of any of the databases to be loaded, irrespective of the needs of the smaller files. If we had a large file of a thousand records of 20 fields and a small look-up table with 10 records of two fields, both would have to be held in identically dimensioned arrays, with most of the second array consisting of empty strings, and a great waste of memory. It would always be necessary to load the largest file first, to ensure the arrays are made large enough.

The alternative approach is to have a separate array for each database so that each can be dimensioned exactly to the right size. This approach is the one used in commercial database systems and is the one we will use in DBQL. Each array will have a different name – a$, b$, c$ . . . etc – and there will have to be a selection

mechanism to call the array which is to be in use at any one time. Superbasic has the SELect ON . . . End SELect structure for this purpose. In commercial programs each database is usually given an 'alias' name by which it can be addressed and which is different from its file name. Superbasic can only SELect on numbers, unfortunately, so each of our files will be known by a number and this will be stored in the variable *alias*.

We need to arrange the program so that, though several different files are held in memory, the program behaves as if, at any one moment, only one file is present and the others remain unseen. Each file will have its associated variables such as nn, fields, chars, etc and each can have associated index files too. Bearing in mind that the procedures of DBQL are written for an array called a$ how can they work for b$ or c$?

Alias and variables

The solution to these problems is simple in principle. Each array will have its own set of variables, so that a$ has ann, afields, achars etc and b$ jas bnn, bfields, bchars and so on. The number of the file in use is held in alias and the various procedures will each have a new introductory procedure to pass the appropriate variables to the procedure proper (see the *QL User Guide* on passing variables). As an example let's look at the changes to the procedure AM (line 2430 in the listing). Please note that the program has been renumbered.

Calling AM from the keyboard now enters the SELect structure and if the value of alias is, say, 1 then the new procedure AMEND is called and has passed to it the array a$ and all the associated variables listed as parameters in line 2450. If alias = 2 then b$ and its variables would be passed. These are the variables that will be needed by AMEND and by the other procedures which will be called by AMEND as it runs. AMEND itself starts at 2500 and consists of the lines which previously made up the old procedure AM in the flat-file version of DBQL. It lists as formal variables in 2450, and must be listed in the same order. The values of some of these variables, of course, are changed as the procedure runs but the new values are automatically passed back to the actual parameter variables.

You will see in the **listing one** that all of the old procedures have been changed in a similar way except for those that are always called by another procedure. This is because variables passed to a procedure are automatically passed on to any other procedure called by the first one. The short procedures DEL and G are too simple to need subsidiary procedures and operate with the actual parameter variables only. Provision is made for up to three files to be held at once.

Procedure LLOAD is especially com-

```
1610 update_index
1620 IF nn=DIMN(x$)THEN PRINT"Saving to microdrive. Please wait.":EXIT
entryloop
1630 END REPeat entryloop
1640 END DEFine enter
1650 REMark ********************** SEARCH
1660 DEFine PROCedure fi
1670 c=0:found=0:CLS
1680 INPUT\"Enter item for search ";s$
1690 SELect ON alias
1700    =1:ac=0:more a$,ann,afields,ac
1710    =2:bc=0:more b$,bnn,bfields,bc
1720    =3:cc=0:more c$,cnn,cfields,cc
1730 END SELect
1740 END DEFine fi
1750 REMark ************** CONTINUE SEARCH
1760 DEFine PROCedure mo
1770 SELect ON alias
1780    =1:more a$,ann,afields,ac
1790    =2:more b$,bnn,bfields,bc
1800    =3:more c$,cnn,cfields,cc
1810 END SELect
1820 END DEFine mo
1830 REMark ************************* MORE
1840 DEFine PROCedure more(x$,nn,fields,c)
1850 d=c+1:IF d>nn THEN PRINT\ TO 9;:PAPER 2:PRINT" end of file ":PAPER
0:RETurn
1860 instr_sr
1870 IF c=nn+1 THEN :IF found THEN :PRINT "NO FURTHER RECORDS
FOUND":ELSE :PRINT\"NOT FOUND":END IF :RETurn :END IF
1880 display x$,fields,c
1890 END DEFine more
1900 REMark ********************** INSTR_SR
1910 DEFine PROCedure instr_sr
1920 c=nn+1
1930 FOR record=d TO nn
1940   FOR fld= 1 TO fields
1950     IF s$ INSTR x$(record,fld) AND x$(record,0,2)<>"S" THEN
c=record:found=1:EXIT record
1960   END FOR fld
1970 END FOR record
1980 END DEFine instr_sr
1990 REMark ********************** DELETE
2000 DEFine PROCedure del
2010 SELect ON alias
2020   =1:a$(ac,0,1)="D":adeleted=adeleted+1
2030   =2:b$(bc,0,1)="D":bdeleted=bdeleted+1
2040   =3:c$(cc,0,1)="D":cdeleted=cdeleted+1
2050 END SELect
2060 PRINT\ "This record is marked for deletion"
2070 END DEFine del
2080 REMark ********************** PRINTER
2090 DEFine PROCedure pr
2100 lprint=1-lprint:menu
2110 END DEFine pr
2120 REMark ************************ GOTO
2130 DEFine PROCedure g(x)
2140 SELect ON alias
2150    =1:ac=x:IF ac>ann THEN ac=ann:END IF :display a$,afields,ac
2160    =2:bc=x:IF bc>bnn THEN bc=cnn:END IF :display b$,bfields,bc
2170    =3:cc=x:IF cc>cnn THEN cc=cnn:END IF :display c$,cfields,cc
2180 END SELect
2190 END DEFine g
2200 REMark ************************ NEXT
2210 DEFine PROCedure n
2220 SELect ON alias
2230   =1:nxt ac,ann:display a$,afields,ac
2240   =2:nxt bc,bnn:display b$,bfields,bc
2250   =3:nxt cc,cnn:display c$,cfields,cc
2260 END SELect
2270 END DEFine n
2280 DEFine PROCedure nxt(c,nn)
2290 c=c+1:IF c>nn THEN PRINT\ TO 30;"end of file":PAUSE 50:c=nn
2300 END DEFine nxt
2310 REMark ************************ BACK
2320 DEFine PROCedure b
2330 SELect ON alias
2340   =1:back ac,ann:display a$,afields,ac
2350   =2:back bc,bnn:display b$,bfields,bc
2360   =3:back cc,cnn:display c$,cfields,cc
2370 END SELect
2380 END DEFine b
2390 DEFine PROCedure back(c,nn)
2400 c=c-1:IF c<1 THEN PRINT\ TO 30;"start of file":PAUSE 50:c=1
2410 END DEFine back
2420 REMark ********************** AMEND
2430 DEFine PROCedure am
2440 SELect ON alias
2450   =1:amend a$,ann,afields,achars,aresave,adeleted,ac,
aindices,andx$
2460   =2:amend b$,bnn,bfields,bchars,bresave,bdeleted,bc,
bindices,bndx$
```

```
2470    =3:amend c$,cnn,cfields,cchars,cresave,cdeleted,cc,
cindices,cndx$
2480 END SELect
2490 END DEFine am
2500 DEFine PROCedure
amend(x$,nn,fields,chars,resave,deleted,c,indices,ndx$)
2510 CLS:CLS#4
2520 PRINT#4,"                        ENTER leaves item unchanged "
2530 PRINT#4,"                     To change an item, type new version
2540 DIM hold$(fields,chars)
2550 resave=1:reenter=0
2560 FOR fld=1 TO fields
2570    PRINT x$(0,fld,4 TO LEN(x$(0,fld)));TO 11;x$(0,fld,3);":
"&x$(c,fld)
2580 PRINT TO 11;
2590 IF x$(0,fld,3)="N" THEN
2600     q$=intake$
2610 ELSE
2620 IF x$(0,fld,3)="D" THEN
2630     q$=checkdate$
2635     q$=date_rev$(q$)
2640 ELSE
2650     INPUT q$
2660 END IF :END IF
2670    IF q$="" THEN :hold$(fld)=x$(c,fld):ELSE
:hold$(fld)=q$:reenter=reenter+(x$(0,fld,1)="I"):END IF
2680 END FOR fld
2690 IF reenter THEN
2700    x$(c,0)="D":deleted=deleted+1:nn=nn+1
2710    FOR j=1 TO fields:x$(nn,j)=hold$(j)
2720    update_index
2730 ELSE
2740    FOR j=1 TO fields:x$(c,j)=hold$(j)
2750 END IF
2760 IF reenter THEN c=nn
2770 display x$,fields,c:menu
2780    IF nn=DIMN(x$) THEN PRINT"Saving file to microdrive. Please
wait.":ssave:lload
2790 END DEFine amend
2800 REMark ************************ QUIT
2810 DEFine PROCedure quit
2820 FOR j=1 TO databases
2830     alias=j
2840     ssave
2850 END FOR j
2860 ending
2870 END DEFine quit
2880 REMark ************************ SAVE
2890 DEFine PROCedure ssave
2900 SELect ON alias
2910    =1:saver a$,ann,afields,achars,adeleted,apac,aoldfile,af$,
aresave,aindices,andx$
2920    =2:saver b$,bnn,bfields,bchars,bdeleted,bpac,boldfile,bf$,
bresave,bindices,bndx$
2930    =3:saver c$,cnn,cfields,cchars,cdeleted,cpac,coldfile,cf$,
cresave,cindices,cndx$
2940 END SELect
2950 END DEFine ssave
2960 DEFine PROCedure saver(x$,nn,fields,chars,deleted,pac,oldfile,f$
resave,indices,ndx$)
2970 IF NOT resave THEN RETurn
2980 dbsave:indexsave:resave=0
2990 END DEFine saver
3000 DEFine PROCedure dbsave
3010 IF oldfile THEN COPY "mdv1_"&f$ TO "mdv1_"&f$&"_temp":DELETE
"mdv1_"&f$
3020 OPEN_NEW #6,"mdv1_"&f$
3030 PRINT#6,nn-(deleted*(pac=1)):PRINT#6,fields:PRINT#6,chars:
PRINT#6,deleted*(1-pac):PRINT#6,indices
3040 FOR record=0 TO nn
3050 IF x$(record,0,1)="D" AND pac=1 THEN NEXT record
3060 FOR fld=0 TO fields
3070 PRINT#6,x$(record,fld)
3080 END FOR fld:END FOR record
3090 CLOSE#6
3100 DELETE "mdv1_"&f$&"_temp"
3110 END DEFine dbsave
3120 DEFine PROCedure indexsave
3130   FOR j=1 TO fields
3140     IF x$(0,j,1)="I" THEN
3150     IF NOT pac THEN COPY "mdv1_"&x$(0,j,4 TO LEN(x$(0,j)))&"_ind"
TO "mdv1_"&x$(0,j,4 TO LEN(x$(0,j)))&"_ind_temp"
3160     DELETE "mdv1_"&x$(0,j,4 TO LEN(x$(0,j)))&"_ind"
3170     IF pac THEN NEXT j
3180     OPEN_NEW#6, "mdv1_"&x$(0,j,4 TO LEN(x$(0,j)))&"_ind"
3190     FOR k=0 TO nn:PRINT#6, ndx$(x$(0,j,2),k,1):PRINT#6,
ndx$(x$(0,j,2),k,2)
3200     CLOSE#6
3210     DELETE "mdv1_"&x$(0,j,4 TO LEN(x$(0,j)))&"_ind_temp"
3220     END IF
3230   END FOR j
3240 END DEFine indexsave
```

plicated because of the fact that an array can only be dimensioned as an actual variable and not in the guise of a formal parameter. This means that LLOAD has to be split into two procedures. SET_ARRAY opens the file and reads in the numerical variables and the program then returns to the calling line to have the main array dimensioned before passing the variables on to INLOAD, which loads the data into the array. If any index files are present the appropriate ndx$ has to be dimensioned at the same time as the database array so the value of *indices* has to be known at this point. Indices is therefore now saved in the file with the other variables, although the older lines which derived indices from the field names have been left as a check. To make database files compatible with both versions of DBQL it is recommended that previous versions be modified to save indices to microdrive.

The procedure START has been changed to initialise the new database variables so that they can be used as passing parameters. Two new global variables have also been added – alias, which has been explained, and *databases*, which stores the number of files currently in memory. It begins with a value of 0 and is incremented whenever a database is loaded or created. Alias begins with a value of 1 and to load the first file call OP in the usual way. To load another file, set alias to another value, up to three, and call OP again. The value of alias is changed by a new procedure USE, passing the required value from the keyboard as USE 2 or USE 3 etc. another way is just to type 'alias=x' <ENTER>, where x is the new value. If you wish to replace one file with another, set alias to the required number and OPen the new file, which will automatically replace the existing one.

As listed, the maximum number of files that can be loaded at once is three. Commercial databases usually allow more files eg *dBase IV* takes up to 10 and some users need still more. The number of files in DBQL can be increased quite easily and the only real limit is the amount of memory available. To add a fourth file, which would be called d$, it would be necessary to add a new line to each of the SElect structures:

=4: [procedure name] d$,dnn,dfields. . . etc,

and list all the d- variables corresponding to the a-, b- and c- variables in the existing lines. A new line would similarly be required in START to initialise the new variables, and the maximum value allowed by USE should be increased. This process could be repeated as often as desired, using the next letter of the alphabet each time as a name for the new array.

The program has been completely renumbered to accommodate the new lines. Line numbers in DBQL are only needed for the programmer to find the way around and are not used by the program itself. A

number of other additions have been made.

SELEKT has been altered to enable 'greater than ' and 'less than' criteria to be used in selections. The new SELEKT, after printing instructions in #4 and displaying the list of field numbers and names, requests an entry in the form of an unbroken string consisting of field number, operator and selection item. The string is then analysed by lines 4270 to 4340. The loop in 4270 to 4290 examines each character in the string until it finds a non-numeric character. If the first character is non-numeric then the string has been incorrectly entered and 4300 prints an error message and loops back to the input.

Line 4310 takes the characters before the first non-numeric character to be the field number and assigns this to *fldno*. The non-numeric character itself is taken to be the operator and assigned to operator$, while the rest of the string is taken to be the selection item and assigned to selct$. Line 4320 checks the validity of the operator and fieldnumber and if one of these is wrong the repeat loop returns to the input with an error message.

If all is well here, the selection loop starting at 4370 is entered, after a numerical value has been obtained from the ASCII code of the operator character in 4350. The loop examines the specified field of each record in the database using a SELect structure to choose the appropriate operator and if a positive result is found it increments the counter variable *selected*. Otherwise a 'S' is attached to the end of field 0 of the record, which is a change from the previous arrangement (see below). At the end of the loop the number of records selected is displayed as before and the same options of displaying or saving the selected records are offered.

The deselection procedure DES has also been changed. Experience has shown that it is often useful to be able to cancel only the last stage of a multiple selection to allow a new selection to be based on a previous one. For instance, a mail order firm, having selected the names of male customers aged under 30, say, might want to select those living in Birmingham and then those living in Brighton or Newcastle in turn, without having to cancel the whole selection and begin anew each time. This is done by adding an 'S' to field 0 whenever a record is selected out, so that a record which has failed three selections would have SSS in its 0 field (after the first character which is reserved for 'D') while one which has only failed the last selection would only have one S.

To go back to the state existing before the latest selection therefore, the last S has to be removed from each record containing Ss, and this option is now offered by DESLCT. Line 4710 offers the choice of complete cancellation as before by inputting a C (upper or lower case) or of cancelling only the last selection by entering S. If C is entered, the loop in 4370 removes all Ss by setting field 0 in each record equal to

```
3250 REMark ************************ OPEN
3260 DEFine PROCedure op
3270 INPUT "Enter file name: ";q$
3280 SELect ON alias
3290   =1:af$=q$:lload:legend af$,ann,aindices
3300   =2:bf$=q$:lload:legend bf$,bnn,bindices
3310   =3:cf$=q$:lload:legend cf$,cnn,cindices
3320 END SELect
3330 databases=databases+1-(databases=3)
3340 END DEFine op
3350 DEFine PROCedure lload
3360 SELect ON alias
3370   =1:set_array af$,ann,afields,achars,adeleted,aindices:DIM
a$(ann+10,afields,achars):DIM andx$(aindices,ann+10,2,achars):inload
a$,ann,afields,aindices,apac,andx$,aoldfile
3380   =2:set_array bf$,bnn,bfields,bchars,bdeleted,bindices:DIM
b$(bnn+10,bfields,bchars):DIM bndx$(bindices,bnn+10,2,bchars):inload
b$,bnn,bfields,bindices,bpac,bndx$,boldfile
3390   =3:set_array cf$,cnn,cfields,cchars,cdeleted,cindices:DIM
c$(cnn+10,cfields,cchars):DIM cndx$(cindices,cnn+10,2,cchars):inload
c$,cnn,cfields,cindices,cpac,cndx$,coldfile
3400 END SELect
3410 END DEFine lload
3420 DEFine PROCedure legend(f$,nn,indices)
3430 CLS:AT 10,7:PRINT"The file '"&f$&"' is open":PRINT\ TO 13;nn;"
records"
3440 IF indices THEN PRINT\ TO 13;indices;" index file(s)"
3450 END DEFine legend
3460 REMark ********************* SET ARRAY
3470 DEFine PROCedure set_array(f$,nn,fields,chars,deleted,indices)
3480 OPEN_IN #6,"mdv1_"&f$
3490 INPUT#6,nn:INPUT#6,fields:INPUT#6,chars:INPUT#6,deleted:
INPUT#6,indices
3500 END DEFine set_array
3510 REMark ********************** INLOAD
3520 DEFine PROCedure inload(x$,nn,fields,indices,pac,xndx$,oldfile)
3530 FOR record=0 TO nn
3540 FOR fld=0 TO fields
3550 INPUT#6,x$(record,fld)
3560 END FOR fld
3570 END FOR record
3580 CLOSE #6:oldfile=1
3590 IF pac THEN RETurn
3600 indices=0:FOR j=1 TO fields:IF x$(0,j,1)="I" THEN
indices=indices+1:x$(0,j,2)=indices
3610 IF NOT indices THEN RETurn
3620 FOR j=1 TO fields
3630   IF x$(0,j,1)="I" THEN
3640     OPEN_IN #6, "mdv1_"&x$(0,j,4 TO LEN(x$(0,j)))&"_ind"
3650     FOR k=1 TO nn:INPUT#6, xndx$(x$(0,j,2),k,1):INPUT#6,
xndx$(x$(0,j,2),k,2)
3660     CLOSE#6
3670   END IF
3680 END FOR j
3690 END DEFine inload
3700 REMark ********************** CREATE
3710 DEFine PROCedure cr
3720 IF alias-databases<1 THEN alias=alias+1-(alias=3)
3730 CLS
3740 INPUT"Please enter file name: ";f$
3750 x$="":fields=0
3760 CLS#4: PRINT#4,"Enter field names in order, up to 8 characters in
length."\"Enter Shift/ESC to quit."
3770 PRINT"    Field name          Character, Number or Date"
3780 REPeat fname
3790 INPUT\ fields+1;" : "; field$;
3800 IF field$="©" THEN EXIT fname
3810 INPUT TO 28;type$;:type$=upper$(type$(1))
3820 IF NOT type$ INSTR "CND" THEN INPUT" Invalid type. Only C,N or D
";type$
3830 IF LEN(field$)>8 THEN field$=field$(1 TO 8)
3840 field$="  "&type$&field$
3850 x$=x$&field$&",":fields=fields+1
3860 END REPeat fname
3870 INPUT\"How many characters in the longest field?"!chars
3880 IF chars<11 THEN chars=11
3890 DIM z$(0,fields,chars)
3900 FOR j=1 TO fields
3910   slice=," INSTR x$:z$(0,j)=x$(1 TO slice-1)
3920   IF slice<LEN(x$) THEN x$=x$(slice+1 TO)
3930 END FOR j
3940 nn=0:deleted=0:indices=0
3950 OPEN_NEW#6,"mdv1_"&f$
3960 PRINT#6,nn:PRINT#6,fields:PRINT#6,chars:PRINT#6,deleted:
PRINT#6,indices
3970 FOR j=0 TO fields:PRINT#6,z$(0,j)
3980 CLOSE #6
3990 databases=databases+1-(databases=3)
4000 CLS:CLS#0:INPUT\"The file '"&f$&"' is ready. Enter records
now?(Y/N) ";q$
4010 IF q$=="Y" THEN
4020   SELect ON alias
4030   =1:af$=f$:lload:en
```

```
4040     =Z:bf$=I$:lload:en
4050     =3:cf$=f$:lload:en
4060   END SELect
4070 END IF
4075 CLS:menu
4080 END DEFine cr
4090 REMark ********************** ENDING
4100 DEFine PROCedure ending
4110 CLS:CLS#0:AT 10,33:PRINT"FILES CLOSED"\\ TO 27;"To use again,
anter RUN."
4120 END DEFine ending
4130 REMark ********************** SELECT
4140 DEFine PROCedure se
4150 SELect ON alias
4160     =1:selekt a$,ann,afields,aselected,ac
4170     =2:selekt b$,bnn,bfields,bselected,bc
4180     =3:selekt c$,cnn,cfields,cselected,cc
4190 END SELect
4200 END DEFine se
4210 REMark ********************** SELEKT
4220 DEFine PROCedure selekt (x$,nn,fields,selected,c)
4230 AT#4,0,0:PRINT#4,"Enter field number, operator(=,< or >) and
selection item as a continuous stringof characters e.g. 10=bingo
"
4240 CLS:FOR j=1 TO fields:PRINT j,x$(0,j,4 TO LEN(x$(0,j)))
4250 REPeat selection
4260 INPUT q$
4270 FOR j=1 TO LEN(q$)
4280     IF NOT (CODE(q$(j))<58 AND CODE(q$(j))>47) THEN EXIT j
4290 END FOR j
4300 IF j=1 THEN PRINT "Invalid entry. Refer to instructions":NEXT
selection
4310 fldno$=q$(1 TO j-1):fldno=fldno$:operator$=q$(j):selct$=q$(j+1 TO)
4320 IF operator$ INSTR "=<>" AND fldno$<= fields THEN EXIT selection
4330 PRINT"Invalid entry. Refer to instructions"
4340 END REPeat selection
4350 operator= CODE(operator$)
4360 selected=0
4370 FOR j=1 TO nn
4380     SELect ON operator
4390         =61:IF selct$ INSTR x$(j,fldno) AND x$(j,0,2)<>"S" THEN
selected=selected+1:ELSE :x$(j,0)=x$(j,0,1 TO LEN(x$(j,0)))&"S":END IF
4400         =60:IF x$(j,fldno)<selct$ AND x$(j,0,2)<>"S" THEN
selected=selected+1:ELSE :x$(j,0)=x$(j,0,1 TO LEN(x$(j,0)))&"S":END IF
4410         =62:IF x$(j,fldno)>selct$ AND x$(j,0,2)<>"S" THEN
selected=selected+1:ELSE :x$(j,0)=x$(j,0,1 TO LEN(x$(j,0)))&"S":END IF
4420     END SELect
4430 END FOR j
4440 PRINT selected;" records selected"
4450 IF NOT selected THEN menu:RETurn
4460 INPUT"Enter S to save selected records to microdrive"\"      L to
list them on screen"\"Press ENTER to return to menu ";q$
4470 IF q$="S" OR q$="s" THEN
4480     INPUT "Enter file name ";subfile$
4490     OPEN_NEW#6, "mdv1_"&subfile$
4500     PRINT#6,selected: PRINT#6,fields: PRINT#6,chars: PRINT#6,0:
PRINT#6,0
4510     FOR j=0 TO nn
4520     IF x$(j,0,2)="S" THEN NEXT j
4530     FOR k=0 TO fields:PRINT#6,x$(j,k)
4540     END FOR j:CLOSE#6
4550 END IF
4560 IF q$="l" OR q$="L" THEN
4570     CLS:FOR k=1 TO nn:IF x$(k,0,2)<>"S" THEN c=k:b_s_display
4580 END IF
4590 menu
4600 END DEFine selekt
4610 REMark ********************** DESELECT
4620 DEFine PROCedure des
4630 SELect ON alias
4640     =1:deslct a$,ann,aselected
4650     =2:deslct b$,bnn,bselected
4660     =3:deslct c$,cnn,cselected
4670 END SELect
4680 END DEFine des
4690 REMark ********************** DESLCT
4700 DEFine PROCedure deslct(x$, nn, selected)
4710 PRINT\ "Enter C to cancel all selections"\"      S to step back
one stage":INPUT opt$
4720 IF opt$="C" OR opt$="c" THEN
4730     FOR j=1 TO nn:x$(j,0)=x$(j,0,1)
4740     selected=0
4750     CLS:PRINT\"All selections cancelled"
4760 ELSE
4770 IF opt$="S" OR opt$="s" THEN
4780     selected=0
4790     FOR j=1 TO nn
4800     IF LEN(x$(j,0))>1 THEN x$(j,0)=x$(j,0,1 TO LEN(x$(j,0))-1)
4810     selected=selected+(LEN(x$(j,0))=1)
4820     END FOR j
4830     IF selected=nn THEN PRINT\"All selections now cancelled":ELSE
:PRINT\ "Last selection cancelled"\selected;" records now selected"
4840 ELSE
4850 PRINT\"Deselect abandoned"
4860 END IF :END IF
4870 END DEFine deslct
4880 REMark ********************
4890 DEFine PROCedure li
4900 SELect ON alias
4910     =1:show a$,ann,afields,
4920     =2:show b$,bnn,bfields,
4930     =3:show c$,cnn,cfields,
4940 END SELect
4950 END DEFine li
4960 REMark ********************
4970 DEFine PROCedure show(x$,n
4980 IF NOT selected THEN PRINT
4990 CLS:FOR k=1 TO nn:IF x$(k,
5000 END DEFine show
5010 REMark ********************
5020 DEFine PROCedure display(x
5030 IF chan=1 THEN CLS
5040 PRINT"record ";c;:IF x$(c,
:PRINT:PRINT:END IF
5050 FOR fld=1 TO fields
5060     IF NOT nemo THEN PRINT#
11;"| ",
5065     IF x$(0,fld,3)="D" THEN
date_rev$(temp$):NEXT fld
5070     PRINT#chan," "&x$(c,fld
5080 END FOR fld
5090 IF lprint AND chan=1 THEN
5100 chan=1
5110 END DEFine display
5120 :
5130 REMark ********************
5140 DEFine PROCedure in
5150 SELect ON alias
5160     =1:index a$,afields,ach
aoldfile,af$
5170     =2:index b$,bfields,bch
boldfile,bf$
5180     =3:index c$,cfields,cch
coldfile,cf$
5190 END SELect
5200 END DEFine in
5210 DEFine PROCedure index(x$,
deleted,oldfile,f$)
5220 DIM ndx$(1,nn,2,chars):act
5230 IF NOT pac THEN
5240     CLS:FOR j=1 TO fields:P
5250     PRINT"Enter field numbe
fldno THEN PRINT"Terminating":R
5260     PRINT"Creating index fi
LEN(x$(0,fldno)))&"_ind"
5270 END IF
5280 x$(0,fldno,1)="I"
5290 FOR k=1 TO nn:ndx$(active,
5300 DIM comp$(2,chars)
5310 FOR item=2 TO nn
5320 in_sort
5330 END FOR item
5340 indname$=x$(0,fldno,4 TO L
5350 DELETE "mdv1_"&indname$
5360 OPEN_NEW#5,"mdv1_"& indnam
5370 FOR k=1 TO nn:PRINT#5,ndx$
5380 CLOSE#5
5390 indices=indices+1
5400 resave=1
5410 dbsave
5420 END DEFine index
5430 REMark ********************
5440 DEFine PROCedure in_sort
5450     p=item
5460     comp$(1)=ndx$(active,p,
5470     ndx$(active,0,1)=comp$(
5480     REPeat compare
5490       IF comp$(1)>ndx$(act
5500
ndx$(active,p,1)=ndx$(active,p-
5510       p=p-1
5520     END REPeat compare
5530     ndx$(active,p,1)=comp$(
5540 END DEFine in_sort
5550 REMark ********** BI
5560 DEFine PROCedure lo
5570 CLS:INPUT\"Please enter i
5580 SELect ON alias
5590     =1:b_search a$,afields,
5600     =2:b_search b$,bfields,
5610     =3:b_search c$,cfields,
5620 END SELect
5630 END DEFine lo
5640 DEFine PROCedure b_search(
```

```
*** LIST

elected,ac
elected,bc
elected,cc

******* SHOW
fields,selected,c)
NO SELECTION HAS BEEN MADE":RETurn
2)<>"S" THEN c=k:b_s_display

**** DISPLAY
fields,c)

1)="D" THEN PRINT"   ( D )":PRINT:ELSE

sn,x$(0,fld,4 TO LEN(x$(0,fld))+1);TO

emp$=x$(c,fld):PRINT#chan,

an=3:display x$,fields,c

 INDEX FILE

s,ann,apac,aresave,aindices,adeleted,

s,bnn,bpac,bresave,bindices,bdeleted,

s,cnn,cpac,cresave,cindices,cdeleted,

elds,chars,nn,pac,resave,indices,

e =1

NT j,x$(0,j,4 TO)
 ";:fldno=getnumber$(0,fields) :IF NOT
urn
";x$(0,fldno,4 TO

1)=v$(k,fldno):ndx$(active,k,2)=k

(x$(0,fldno))&" ind"

ctive,k,1):PRINT#5,ndx$(active,k,2)

* IN SORT

):comp$(2)=ndx$(active,p,2)
):ndx$(active,0,2)=comp$(2)

<,p-1,1) THEN EXIT compare

,)  ndx$(active,p,2)=ndx$(active,p-1,2)

):ndx$(active,p,2)=comp$(2)

ARY SEARCH

 for look for: ";s$

nn,ac,aactive;andx$,aindices
nn,bc,bactive,bndx$,bindices
nn,cc,cactive,cndx$,cindices

s,fields,nn,c,active,ndx$,indices)
```

```
5550 REMark ***************** BINARY SEARCH
5560 DEFine PROCedure lo
5570 CLS:INPUT\"Please enter item to look for: ";s$
5580 SELect ON alias
5590    =1:b_search a$,afields,ann,ac,aactive,andx$,aindices
5600    =2:b_search b$,bfields,bnn,bc,bactive,bndx$,bindices
5610    =3:b_search c$,cfields,cnn,cc,cactive,cndx$,cindices
5620 END SELect
5630 END DEFine lo
5640 DEFine PROCedure b_search(x$,fields,nn,c,active,ndx$,indices)
5650 FOR j=1 TO fields:IF x$(0,j,1)="I" THEN PRINT x$(0,j,2),x$(0,j,4
TO)
5660 PRINT\"Current field for search is ";active\"Press ENTER to
confirm "\"or enter another number:  ";: q$=getnumber$(0,indices): IF
q$<>"0" THEN active=q$
5670 FOR sfld= 1 TO fields:IF x$(0,sfld,1 TO 2)="I"&active THEN EXIT
sfld
5680 z=bin_s(nn,s$,ndx$)
5690 c=ndx$(active,z,2)
5700 IF x$(c,sfld,1 TO LEN(s$))<>s$ THEN PRINT\"NOT FOUND ":RETurn
5710 CLS
5720 updown ndx$,z
5730 END DEFine b_search
5740 REMark ***************** FN BIN SEARCH
5750 DEFine FuNction bin_s(n,x$,y$)
5760 LOCal s
5770 power=INT(LN(n-1)/LN(2))
5780 s=2^power
5790 FOR k=power-1 TO 0 STEP -1
5800 s=s+(2^k)*(x$>y$(active,s,1,1 TO
LEN(x$)))-(2^k)*(x$<y$(active,s,1,1 TO LEN(x$)))
5810 IF s>n THEN s=n
5820 IF s<1 THEN s=1
5830 END FOR k
5840 RETurn s
5850 END DEFine bin_s
5860 REMark ********************** UPDOWN
5870 DEFine PROCedure updown(y$,s)
5880 REPeat upward
5890 s=s-1
5900 IF s=0 THEN EXIT upward
5910 c=y$(active,s,2)
5920 IF x$(c,sfld,1 TO LEN(s$))<>s$ THEN EXIT upward
5930 END REPeat upward
5940 REPeat downward
5950 s=s+1
5960 c=y$(active,s,2)
5970 IF x$(c,sfld,1 TO LEN(s$))<>s$ THEN EXIT downward
5980 b_s_display
5990 IF s=nn THEN EXIT downward
6000 END REPeat downward
6010 END DEFine updown
6020  REMark ***************** B S DISPLAY
6030 DEFine PROCedure b_s_display
6040 PRINT#chan, "# ";c&"  ";
6042 FOR j=1 to fields
6044  IF x$(0,j,3)="D" THEN temp$=x$(c,j):PRINT#chan, date_rev$(temp$)
: NEXT j
6046  PRINT#chan, x$(c,j)&" ";
6048 END FOR j
6049 PRINT#chan
6050 IF lprint AND chan=1 THEN chan=3:b_s_display
6060 chan=1
6070 END DEFine b_s_display
6080 REMark ***************** UPDATE_INDEX
6090 DEFine PROCedure update_index
6100 IF NOT indices THEN RETurn
6110 DIM comp$(2,chars)
6120 FOR j=1 TO fields
6130   IF x$(0,j,1)="I" THEN
6140     active=x$(0,j,2)
6150     ndx$(active,nn,1)=x$(nn,j):ndx$(active,nn,2)=nn:item=nn
6160     in_sort
6170   END IF
6180 END FOR j
6190 END DEFine update_index
6200 REMark ********************** PACK
6210 DEFine PROCedure pa
6220 SELect ON alias
6230    =1:pak a$,afields,achars,ann,adeleted,aindices,apac,aresave
6240    =2:pak b$,bfields,bchars,bnn,bdeleted,bindices,bpac,bresave
6250    =3:pak c$,cfields,cchars,cnn,cdeleted,cindices,cpac,cresave
6260 END SELect
6270 END DEFine pa
6280 DEFine PROCedure
pak(x$,fields,chars,nn,deleted,indices,pac,resave)
6290 pac=1:resave=1
6300 ssave
6310 lload
6320 CLS:PRINT\"Rebuilding the index files.  Please wait."
6330 FOR j=1 TO fields
6340    IF x$(0,j,1)="I" THEN
6350      fldno=j
6360      PRINT\x$(0,j,4 TO LEN(x$(0,j)))&" index"
6370      in
6380    END IF
6390 END FOR j
6400 pac=0:resave=0
6410 lload
6420 END DEFine pa
```

```
6430 REMark ************************** USE
6440 DEFine PROCedure use(x)
6450 IF x>3 THEN x=3
6460 alias=x
6470 END DEFine use
6480 REMark ********************** DATE_REVERSE
6490 DEFine FuNction date_rev$ (x$)
6500 slice1$=x$(1 TO 2):slice2$=x$(3 TO 4):slice3$=x$(5 TO 6)
6510 x$=slice3$&slice2$&slice1$
6520 RETurn x$
6530 END DEFine date_rev$
```

its first character only. If option S is entered, field 0 of each record is shortened by removing the last character, provided its length is already more than one. A field of only one character length cannot contain any Ss and is left unchanged. Appropriate messages are printed to screen with each option. Note also that 1600 in ENTER now inserts only one blank character in field 0 in a new record.

If dates are to be selected on using > or < operators then the previous format of date strings is unsuitable, since years are more important than days, and the order of day, month and year needs to be reversed. This is done by a new function DATE_REV$ (line 6490 onwards), which slices the date string into three separate sections for day, month and year and then recombines them in reverse order. The function works equally well for restoring a reversed date to normal format for printing. It is recommended that all dates are stored in reverse form and 1520 reverses a newly entered date in ENTER, so that keyboard entry remains as before, while 2635 does the same for AMEND. Line 5065 prints a reversed (ie normal) date in DISPLAY and line 6044 in B_S_DISPLAY.

CReate now offers the option of entering data immediately into a newly created database. If this option is accepted, the empty file is loaded according to the current setting of *alias* and EN is called. QUIT has been altered to check all the databases in memory and call SSAVE for each in turn.

The changes in SSAVE have made it impossible to address DBSAVE directly and PA(ck) now has to call SSAVE. It is therefore necessary to set and cancel the flag *resave* at the same time as *pac*. These statements have been added to PA.

Facilities for printing output on paper have so far been limited and the system has been radically revised. PR now toggles the value of a global flag *print* which is initially set to 0 by START. DISPLAY has been changed so that when 1 print is set and chan = 1 line 5090 sets chan to 3 and then calls DISPLAY itself – a form of looping – and on the second time round, the output goes to channel 3, the printer. With chan set at 3, line 5090 is skipped on the second pass, (avoiding a perpetual loop!) and 5100 resets chan to 1. The same system has been installed in B_S_ DISPLAY so that results of selections and binary searches can now be printed out. The effect is that with print set, any data printed to screen is echoed to the printer. A new line in MENU now prints a message 'Is the printer ready?' in channel 4 whenever 1 print is set. If the printer is not ready the program will 'hang', waiting for the printer, and this condition could be mistaken for a 'crash'.

Two modifications have been made to ENter. The previous method of saving and reloading the file after entering 10 records no longer works with the new SELect structures and instead the whole procedure has been enclosed in a repeat loop — *entryloop* — which has the same effect. Secondly the display of field names has been changed by the addition of the nested FOR — NEXT loops j and k in 1390 and 1400 so that all the field names are displayed at once, instead of one by one, before input begins with the first field. If the database has more than 19 fields (the number of lines in window #1) the first 19 are displayed, followed by the remainder when the first 19 have been filled.

A full listing of the amended program is included in this article except for the validation functions INTAKE$, CHECKDATE$, UPPER$, LOWER$ and GETNUMBER$. These are exactly the same as in last month's listing, except for new line numbers, and should be added at the end of this listing.

Next month I will deal with programming the database and tailoring it for specific applications.

aCcLcesBs

BELGIUM

Club Sinclair BruQsL (Belgium) Contact: Jaques Tasset, Aarlenstraat 104, 1040 Brussels, Belgium

SWEDEN

International QL Conference bulletin board system (Swedish and English). Contact: Michael Cronsten, System Operator, Jamten-TCL, S Soere 1073, 83030 Lit, Sweden.

USA

New England Sinclair QL User Group (USA) Membership Secretary: Sherm Waterman, PO Box 8763, Boston, MA 02114 8763, USA. Magazine: *NESQLUG News*. Editor: Peter Hale, 195 Central Ave., Chelsea, MA 02150, USA.

NORWAY

Norweigan All Sinclair Association (NASA) Contact: P Monstad, NASA, N-5580 Oelen, Norway. Magazine: *Sinclair Magazine*.

ITALY

Qitaly Club Chairman: Roberto Orlandi, Via Brescia 26, 25039 Travagliato (BS), Italy. Tel. (local) +39 30 6863311. Magazine: *Qitaly Magazine*. Editor: Dr Eros Forenzi, Via Valeriana 44, 23010 Berbenno (SO), Italy. Tel. (local) +39 342 492323.

TURKEY

QL Qlub (Turkey). Contact: Bulent Artuz, Prof. Sitesi B/1 D/5, Etiler 80600, Istanbul, Turkey.

ENGLAND

Quanta (UK) Membership Secretary, Bill Newell, 213 Manor Road, Benfleet, Essex SS7 4JD. Magazine: *Quanta*. Editor: Bill Fuggle, 20 Widnes Avenue, Selly Oak, Birmingham B29 6QE.

Bristol sub-group: Roy Brereton, 94 Teignmouth Rd, Clevedon, Avon BS21 6DR.

Essex sub-group: Bob Fingell, 22 Paley Gardens, Loughton, Essex IG10 2AN.

London sub-group: Jeremy Davis, 6 Elmcroft Crescent, Harrow, Middlesex HA2 6HN.

Northern Ireland: Billy Turkington, Fairyhill, Rostrevor, Newry, Co., Down BT34 3BB.

Qubbesoft PD library. Contact: Ron Dunnett, 38 Brunwin Rd, Rayne, Braintree, Essex CM7 5BU.

SCOTLAND

Scottish QL Users Group Contact: Alan Pemberton, 65 Lingerwood Rd., Newtongrange, Midlothian EH22 4QQ. Newsletter.

HOLLAND

Sin_QL_Air (Netherlands) Membership Secretary: Bob Visser, Snelrewaard 6, 2904 SN Capelle, a/d Ijssel, Netherlands. Magazine: *Quasar*. Editor: C H M Biemans, Elzenstraat 5, 5461 CL Veghel, Netherlands.

GERMANY

Sinclair QL User Club eV (Germany) Foreign Contact: Franz Herrmann, Talstrasse 21, d-W5460 Ochenfels, West Germany. Magazine: *Quasar*.

SEIKOSHA DOT MATRIX PRINTERS

Silica Systems are pleased to recommend Seikosha printers at ***NEW LOW PROMOTIONAL PRICES***. The Seikosha range is built to exacting standards, by a company that is used to manufacturing high quality precision products. In fact, you may find that you are already wearing one of these products on your wrist as Seikosha are part of the massive Seiko/Epson group (with a turnover of £6 billion and over 18,000 staff!).

When you buy a Seikosha printer from Silica, we will give you a Silica Printer Starter Kit (worth £25+VAT = £29.38) FREE OF CHARGE, to make sure you get off to a flying start with your new printer. For further information on the Seikosha range, complete and return the coupon below. Or, if you can't wait to get your hands on a new low priced Seikosha printer, telephone our mail order hotline, 081-309 1111 to place your order now.

- **FREE DELIVERY** Next day - anywhere in the UK mainland
- **FREE STARTER KIT** Worth £29.38 with every Seikosha printer.
- **FREE HELPLINE** Technical helpline during office hours.
- **1 YEAR WARRANTY** With every Seikosha printer from Silica.

Printer speeds are quoted as follows:
SD = Superdraft 15cpi, D = Draft 12cpi, NLQ = Near Letter Quality 12cpi, LQ = Letter Quality 12cpi.

9-PIN PRINTER

192 CPS DRAFT **80** COL
- *Seikosha SP-1900AI - 9 pin - 80 col*
- *192cps Draft, 48cps NLQ*
- *1K Printer Buffer + 2 Fonts*
- *Parallel Interface*
- *Graphics Resolution: 144×72dpi*
- *Epson and IBM Emulation*
- *Paper Parking*
- *FREE Printer Starter Kit*

SP-1900AI RRP	£179
STARTER KIT	£25
TOTAL RRP:	£204
Less Silica Saving:	£105
SILICA PRICE (EXC VAT):	£99

£99
+VAT= £116.33 Ref: PRI 8190

9-PIN PRINTERS

192 CPS DRAFT **80** COL
- *Seikosha SP-2000AI - 9 pin - 80 col*
- *192cps Draft, 48cps NLQ*
- *21K Printer Buffer + 2 Fonts*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 144×72dpi*
- *Epson and IBM Emulation*
- *Optional Auto Sheet Feeder Unit*
- *FREE Silica Printer Starter Kit*

RRP:	£199
STARTER KIT	£25
TOTAL RRP:	£224
SAVING:	£95
SILICA PRICE:	£129

£129
+VAT=£151.58 ref: PRI 8200

240 CPS DRAFT **136** COL
- *Seikosha SP-2415AI - 9 pin - 136 col*
- *300cps SD, 240cps Draft, 50cps NLQ*
- *17.5K Printer Buffer + 5 Fonts*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 240×144dpi*
- *Epson and IBM Emulation*
- *Superior Paper Handling*
- *FREE Silica Printer Starter Kit*

RRP:	£320
STARTER KIT	£25
TOTAL RRP:	£345
SAVING:	£136
SILICA PRICE:	£209

£209
+VAT=£245.58 ref: PRI 8215

24-PIN PRINTERS

240 CPS DRAFT **80** COL
- *Seikosha SL-92AI - 24 pin - 80 col*
- *240cps Draft, 80cps LQ*
- *44K Printer Buffer + 2 Fonts*
- *Parallel Interface*
- *Graphics Resolution: 360×360dpi*
- *Epson Emulation*
- *Optional Font ROM - 7 Fonts*
- *FREE Silica Printer Starter Kit*

RRP:	£335
STARTER KIT	£25
TOTAL RRP:	£360
SAVING:	£171
SILICA PRICE:	£189

£189
+VAT=£222.08 ref: PRI 8292

- *Includes Programmable ROM Card*

324 CPS DRAFT **80** COL
- *Seikosha SL-210AI - 24 pin - 80 col*
- *324cps Draft, 108cps LQ*
- *5K Printer Buffer + 9 Fonts*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 360×360dpi*
- *Epson and IBM Emulation*
- *Optional Auto Sheet Feeder*
- *FREE Silica Printer Starter Kit*

RRP:	£599
STARTER KIT	£25
TOTAL RRP:	£624
SAVING:	£255
SILICA PRICE:	£369

£369
+VAT=£433.58 ref: PRI 8321

- *Handles 1+2 Multipart Forms*
- *Rechargeable Battery Pack Optional*

144 CPS DRAFT **80** COL
- *Seikosha LT-20 - 24 pin - 80 col*
- *180cps SD, 144cps Draft, 60cps LQ*
- *1K Printer Buffer + 9 Fonts*
- *Parallel Interface*
- *Graphics Resolution: 360×180dpi*
- *IBM Emulation*
- *Built-in 50 Sheet Auto Feeder*
- *FREE Silica Printer Starter Kit*

RRP:	£369
STARTER KIT	£25
TOTAL RRP:	£394
SAVING:	£125
SILICA PRICE:	£269

£269
+VAT=£316.08 ref: PRI 8820

BUSINESS PRINTERS (WIDE-BODY)

- *Includes Programmable ROM Card*

277 CPS DRAFT **136** COL
- *Seikosha SL-230AI - 24 pin - 136 col*
- *277cps Draft, 92cps LQ*
- *5K Printer Buffer + 9 Fonts*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 360×360dpi*
- *Epson and IBM Emulation*
- *Optional Auto Sheet Feeder*
- *FREE Silica Printer Starter Kit*

RRP:	£699
STARTER KIT	£25
TOTAL RRP:	£724
SAVING:	£275
SILICA PRICE:	£449

£449
+VAT=£527.58 ref: PRI 8423

462 CPS DRAFT **136** COL
- *Seikosha BP-5500AI - 8 pin - 136 col*
- *462cps Draft, 106cps NLQ*
- *18K Printer Buffer*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 240×144dpi*
- *Epson and IBM Emulation*
- *Optional Auto Sheet Feeder*
- *FREE Silica Printer Starter Kit*

RRP:	£1299
STARTER KIT	£25
TOTAL RRP:	£1324
SAVING:	£325
SILICA PRICE:	£999

£999
+VAT=£1173.83 ref: PRI 8755

- *Low Form Tear-off*

520 CPS DRAFT **136** COL
- *Seikosha BP-5780AI - 18 pin - 136 col*
- *780cps SD, 520cps Draft, 130cps NLQ*
- *20K Printer Buffer + 2 Fonts*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 240×240dpi*
- *Epson and IBM Emulation*
- *Handles 1+6 Multipart Forms*
- *FREE Silica Printer Starter Kit*

RRP:	£1699
STARTER KIT	£25
TOTAL RRP:	£1724
SAVING:	£325
SILICA PRICE:	£1399

£1399
+VAT=£1643.83 ref: PRI 8780

800 CPS DRAFT **136** COL
- *Seikosha SBP-10AI - 18 pin - 136 col*
- *800cps Draft, 200cps NLQ*
- *64K Printer Buffer + 2 Fonts*
- *Parallel and Serial Interfaces*
- *Graphics Resolution: 240×240dpi*
- *Epson and IBM Emulation*
- *Built-in Cut Sheet Feeder*
- *FREE Silica Printer Starter Kit*

RRP:	£2999
STARTER KIT	£25
TOTAL RRP:	£3024
SAVING:	£525
SILICA PRICE:	£2499

£2499
+VAT=£2936.33 ref: PRI 8810

FREE! PRINTER STARTER KIT

Every Seikosha printer from Silica, comes with the Silica Printer Starter Kit, worth £29.38 (£25 +VAT), completely FREE OF CHARGE. It includes all you need to get up and running with your new printer.
- *3½" Disk with Amiga & ST Printer Drivers*
- *1.8 Metre Parallel Printer Cable*
- *200 Sheets of Quality Continuous Paper*
- *100 Continuous Tractor Feed Address Labels*
- *5 Continuous Envelopes on Tractor Feed*

SILICA SYSTEMS OFFER YOU

- ***FREE OVERNIGHT DELIVERY:*** On all hardware orders shipped in the UK mainland.
- ***TECHNICAL SUPPORT HELPLINE:*** Team of PC technical experts at your service.
- ***PRICE MATCH:*** We normally match competitors on a "Same product - Same price" basis.
- ***ESTABLISHED 12 YEARS:*** Proven track record in professional computer sales.
- ***BUSINESS + EDUCATION + GOVERNMENT:*** Volume discounts available for large orders.
- ***SHOWROOMS:*** Demonstration and training facilities at our London & Sidcup branches.
- ***THE FULL STOCK RANGE:*** All of your PC requirements from one supplier.
- ***FREE CATALOGUES:*** Will be mailed to you with offers and software/peripheral details.
- ***PAYMENT:*** By cash, cheque and all major credit cards.

Before you decide when to buy your new printer, we suggest you think very carefully about WHERE you buy it. Consider what it will be like a few months after you have made your purchase, when you may require additional peripherals or software, or help and advice. And, will the company you buy from contact you with details of new products? At Silica Systems, we ensure that you will have nothing to worry about. We have been established for over 12 years and, with our unrivalled experience and expertise, we can now claim to meet our customers requirements with an understanding which is second to none. But don't just take our word for it. Complete and return the coupon now for our latest Free literature and begin to experience the "Silica Systems Service".

MAIL ORDER HOTLINE 081-309 1111

SILICA

MAIL ORDER:	1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX	**Tel: 081-309 1111**	
Order Lines Open:	Mon-Sat 9.00am-6.00pm	No Late Night Opening	Fax No: 081-308 0608
LONDON SHOP:	52 Tottenham Court Road, London, W1P 0BA	**Tel: 071-580 4000**	
Opening Hours:	Mon-Sat 9.30am-6.00pm	No Late Night Opening	Fax No: 071-323 4737
LONDON SHOP:	Selfridges (1st Floor), Oxford Street, London, W1A 1AB	**Tel: 071-629 1234**	
Opening Hours:	Mon-Sat 9.30am-6.00pm	Late Night: Thursday until 8pm	Extension: 3914
SIDCUP SHOP:	1-4 The Mews, Hatherley Rd, Sidcup, Kent, DA14 4DX	**Tel: 081-302 8811**	
Opening Hours:	Mon-Sat 9.00am-5.30pm	Late Night: Friday until 7pm	Fax No: 081-309 0017

To: Silica Systems Ltd, Dept. QLW1091-58B, 1-4 The Mews, Hatherley Rd, Sidcup Kent, DA14 4DX

PLEASE SEND FREE SEIKOSHA INFORMATION

Mr/Mrs/Ms: ..... Initials: ..... Surname: .....

Company Name (if applicable): .....

Address: .....

..... Postcode: .....

Tel (Home): Tel (Work):

LANSTEANT

HARDWARE

Care Electronics
0923 672102
Peripherals and accessories.
Thor International
010 4533 930305
010 4533 937544
Digital Precision
081-527 5493
Discs, interfaces, drives, keyboards, RAM expansions.
EEC Ltd
0753 888866
QLs, peripherals, accessories
Miracle Systems
0904 423986
Trump Card interface, modems, hard disc drive.
Strong Computer Systems
0267 231246
Sector Software
0772 454328
Peripherals.
S.P.E.M
Via Aosta 16, 10154 Torino, Italy.
+39 11 857924 Fax +39 11280009
QL disk interfaces, ram, eprom holders.
TF Services
071 724 9053
Repairs, spares, peripherals.
TK Computerware
0303 812801
Peripherals.

SERVICES

Adman Services
0952 255895
Joe Atkinson
36 Ranelagh Road, London W5 5RJ
Roms, mdvs and all spares.
Suredata
081-902 5218
Repairs.
QL World
(reader services)
Subscriptions
091 510 8787 (Lazerhold)
Microdrive Exchange
0772 454328 (Sector)
Recent back issues : write to editor
Quanta
User Group, support.
Hon. secretary, Phil Borman.
0933 460690
Xchange User Group
Freepost
Beckenham
Kent BR3 2BR

SOFTWARE

Athene Consultants
0705 511439
ARK Distribution
0983 79496
Archivist, Master Spy and other business and utility software.
Call Us What You Like
55 Greenan, Shaws Road, Belfast BT11 8LX. No telephone. Superspy.
Care Electronics
0923 672102
Tony Tebby software. Miracle hardware, peripherals, utilities.
CGH Services
0559384 574
Adventures. QLAF, DIY Toolkit.
Compware
0270 582301
Task Swopper, Mega Toolbox, Expert, System Shell, Speedscreen and others.
Digital Precision
081-527 5493
Turbo Basic compiler, Desktop Publisher, Editor, Eye-Q Graphics. Sprite generator, Super Forth, Better Basic, Professional and Super Astrologer, Media Manager, Supercharge, Blocklands, Driodzone, Aracadia, Reversi, Backgammon, Master Bridge Tutor, C-M Emulator, C compiler, Lightning speed enhancer.
DJW Software
0256 881701
Homebanker
Dilwyn Jones Computing
0248 354023
Home Budget, Vision Mixer, QL Genealogist, Textidy, Spellbound and others.
Eidersoft
089283 2552
Games utilities, Ice. Pyramide.
ImaQLate Software
42 Albion Street, Broadstairs, Kent CT10 1NE.
Biograph.
Jochen Merz Software
010 49 203501274
QL/Atari Emulators, utility and games software.
Lear Data Systems
6 Southview Green, Bentley
Ipswich, Suffolk IP9 2DR
Qtris
Liberation Software
081 546 7795
Basic compiler (QLib) and utilities.
QFILE
Apartado 2110 p-1103 Lisboa Codex, MS-Link and Discopy. Portugal.
QL Supersoft
0256 475992
Spell, Numbers, Superdraw.
Qubbesoft PD
38 Brunwin Road, Rayne, Braintree, Essex CM7 5BU
Public domain software.
Sector Software
0772 454328
Flashback and other utilities.
SD Microsystems
0462 422897
Business software including General Ledger. Small Traders' Pack/Invoicer and Stock Accounting.
Software 87
33 Savernake Road, London NW3 2JU
Text 87 and updates.
Talent Software
0303 813883
Utilities, games.
TK Computerware
0303 812801
Add-ons, spares, software.
WD Software
0534 81392

MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS

Programmieren in QDOS
ca. 400 Seiten DM 79,80
H. - Peter Recktenwald - Tel.: 852 0413
D-1000 Berlin 41, Albestr. 21
Eine ausführliche Beschreibung des Betriebssystems und seiner Arbeitsweise einschließlich Pointer Interface, Thing-Erweiterung und der programmierbaren Hardware. Mit Beispielen zu den gängigen Standards und allen wichtigen Details der Programmierung, Jobs, neuen Devices, Erweiterungen und Parameter-Angaben für Basic. Tabellen mit Kurzbeschreibung zu Systemzahlen, Traps und Vectoren. Beispielprogramme und assemblertaugliche Tabellen und Texte auch auf Discette. Vollständiges Inhaltsverzeichnis gegen Freiumschlag an obige Adresse.

QL REAL TIME DIGITIZER
Digitize moving pictures in real time. Transforms QL into a video printer or 3-D image digitizer via a video camera/recorder
* Digitizes in .02 secs.
* 256x256 resolution in 8 levels
* Full, ¼ or 4x ¼ screen modes
* Screen update - 5 or 2.7 frame/sec
* 1:1 grey level 9-pin printer dump.
* Interfaces to QL ROM PORT
* Runs on unexpanded QL and multitasks.
* Cost: £99.00 incl. State mdv or flp.

£99

CL SYSTEMS, 403 CHAPTER ROAD, DOLLIS HILL LONDON NW2 5NG TEL: 081 459 1351

QL ROMS AND SPARES
ROMS, MGE (Spanish, MGF (French), MGG (German), JS (English), £20 each. JM (English), £10
MDVS. Complete QL MDV Subunit £20, head and chassis £10, Microswitch £2, Motor £6. Rubber & Plastic Roller £2. MDV2 Stand-off spacer + srew £2. ULA £5. Tr 4,5,6,7 set £2. MDV pcb inc, ULA £11. Prices include UK postage. Send SAE for price list of all QL spares.
Joe Atkinson, 36 Ranelagh Road, Ealing, London W5 5RJ (Mail Order Only).

QL BARGAIN. Over £2,000 worth of equipment at original prices.
JM QL,768K Trumpcard, twin 3.5in disk drives, QCom Modem complete with comms software, Q-Power Battery Backup. Colour monitor, Star LC10 dot matrix printer. Most DP main software programs complete with manuals + over thirty other programs on disk and microdrive. Many spare microdrive cartridges. Text books on QL users and magazines. Offers for the whole system only. £700 ono. 081-290 4061 evenings.

QL SYSTEMS Microvitec colour monitor, Users Guide, ARCHIVE Handbook Pyramide's THE DESIGNER and Ultrasoft's QWRITER (microdrive versions) £60 the lot (plus p&p). Telephone (0532) 747705

MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS
MICRO ADS

QTop is the definitive User-Front-End package for all QL compatible computers, the **SINCLAIR QL** itself, all **CST** Thor computers including the powerful Thor **XVI,** the **ATARI ST** and **AMIGA** with QL-Emulators. **QTop** is the QL's equivalent to **WINDOWS** on MS-DOS PC's. The powerful QJUMP extended Pointer Environment is included and if it's installed all its facilities are supported. The main menus of **QTop** are **DESK, DEVICES, FILES, JOBS, KEYBOARD, PROGRAMS, MY PROGS, TOOLS** and **OPTIONS** with over 110 sub-menus. **QTop** is ideal for Gold-Card or Harddisk users as it supports real Sub-Directories, The **QTop** package includes five disktop accessories: **CLOCKS, DEMOS, ANIMATOR, INDEX** and **SNAP.** Software with 41 page manual is available in English or German. Free updates and user support. Ask for free product information sheet. Prices including P&P are SFR 90 (on one 3.5" or 5.25" disc) and SFR 98 (on three mdv). Send orders with cheques drawn in SFR to:

COWO Electronic
Urs König, Münsterstrasse 4, CH-6210 Sursee, Tel 045-211478

or contact your local distributor: **TK Computerware** in the UK and **QLympic Computer Systems** in Germany.

CLASSIFIED

Please send coupon to: Advertising Department Sinclair/QL World, Panini House, 116/120 Goswell Road, London EC1. Pleae place this Classified Advertisement in the next issue of Sinclair/QL World.

NAME .....

ADDRESS .....

.....

DAYTIME TELEPHONE NO .....

I enclose a cheque/P.O. to the value of £17.65 (inc. VAT) for a lineage advertisement of no more than 30 words. Please make cheques payable to MCPC Ltd

SINCLAIR

THE PREMIER MAGAZINE

QL

FOR SINCLAIR QL USERS

For one month only, Sinclair QL World is offering you the chance to subscribe for 12 months and receive 14 issues. Yes, that's 2 free issues! Subscribe now!

The first issue of a new subscription to be delivered will be one or two issues after the one you placed your order in

Please send me one year's subscription to **Sinclair QL World.** I enclose my cheque/money order for £..... made payable to MCPC LIMITED or debit my Access/Visa card.

Card No. | | | | | | | | | | | | | | | |

Expiry date | | | | | | |

Name .....

Address .....

.....

Postcode ..... (Please enter postcode to ensure prompt delivery)

Signed ..... Date..... SQLW1091

Send to: M.S.M. Subs. Dept., Lazahold Ltd., P.O. Box 10, Roper Street, Pallion Ind. Estate, Sunderland SR4 6SN.

U.K.: £23.40, EUROPE: £32.90, REST OF THE WORLD: £40.90

OVERSEAS RATES INCLUDE AIRMAIL SERVICE